NAWCWPNS TP 8206

SMERF
A Two-Dimensional Eulerian Code for Reactive Flow

by

Eric Lundstrom
Naval Air Warfare Center Weapons Division
China Lake, California

and

Larry Libersky
Energetic Materials Research Testing Center
New Mexico Institute of Mining and Technology
Socorro, New Mexico

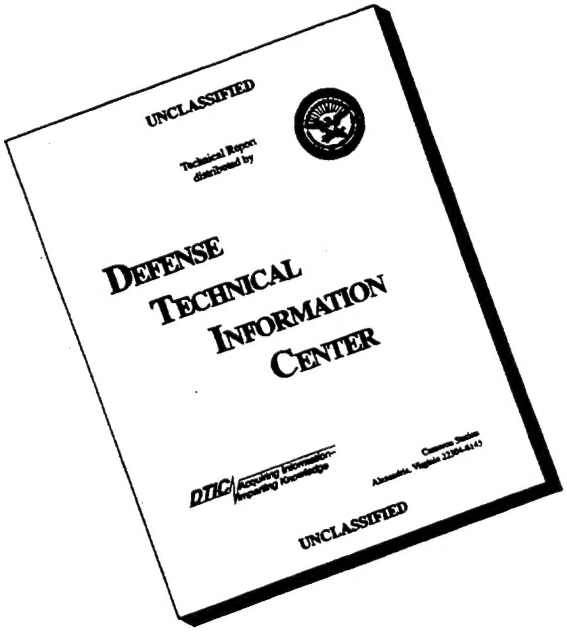
for

Carolyn A. Dettling
Coordinator, IMAD Ordnance Project
Naval Air Warfare Center Weapons Division
China Lake, California

FEBRUARY 1996

NAVAL AIR WARFARE CENTER WEAPONS DIVISION
CHINA LAKE, CA 93555-6001

DISCLAIMER NOTICE



THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

Naval Air Warfare Center Weapons Division

FOREWORD

This report documents the SMERF hydrodynamic code written at the Naval Air Warfare Center Weapons Division, China Lake, Calif., and at the New Mexico Institute of Mining and Technology, Socorro, NM. The code was developed under the sponsorship of the Naval Sea Systems Command Insensitive Munitions Advanced Development Program (Don Porada, Cognizant Technology Manager) under Project Element No. 63609N and Appropriation No. AB1741319.

As modifications continue to be made to the code, this document is released at the working level, and, as such, does not necessarily reflect the views of the Naval Air Warfare Center Weapons Division (NAWCWPNS).

Moreover, because this document is intended as a manual for the SMERF code and was compiled by a contractor, it does not conform in all formatting details to standard NAWCWPNS report requirements.

This report has been reviewed for technical accuracy by Olaf E. R. Heimdahl and Carolyn A. Dettling.

Approved by
D. A. GOSS, *Head*
Ordnance Systems Department
1 February 1996

Under authority of
D. B. McKINNEY
RAdm., U.S. Navy
Commander

Released for publication by
S. HAALAND
Director for Research & Engineering

NAWCWPNS Technical Publication 8206

Published by ..... Scientific and Technical Documentation
Collation ..... Cover, 23 leaves
First printing ..... 400 copies

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	February 1996	Manual

4. TITLE AND SUBTITLE

SMERF—A TWO-DIMENSIONAL EULERIAN CODE FOR REACTIVE FLOW

5. FUNDING NUMBERS

Appropriation No. AB1741319

6. AUTHOR(S)

Eric Lundstrom, Larry Libersky

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Air Warfare Center Weapons Division
China Lake, CA 93555-6001

8. PERFORMING ORGANIZATION REPORT NUMBER

NAWCWPNS TP 8206

8. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Naval Sea Systems Command
Washington, D.C. 20362

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

This manual documents the SMERF hydrodynamic code. SMERF was written at the New Mexico Institute of Mining and Technology, Socorro, NM, and at the Naval Air Warfare Center Weapons Division, China Lake, CA. The code has general applicability, but has been most exercised toward simulation of shock initiation and sympathetic detonation of explosives.

14. SUBJECT TERMS

Insensitive Munitions, SMERF

15. NUMBER OF PAGES

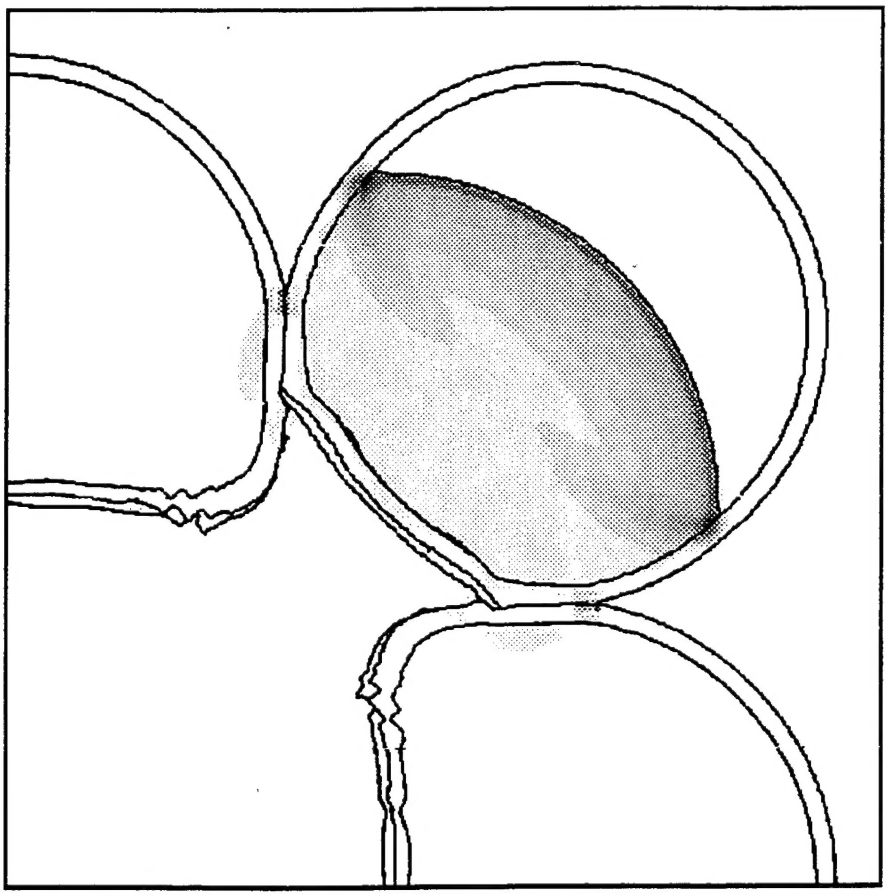
44

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAR

SMERF

A Two-Dimensional Eulerian Code for Reactive Flow



Eric Lundstrom

Larry Libersky

CONTENTS

I. INTRODUCTION

This manual documents the **SMERF** hydrodynamics code. **SMERF** was written at the New Mexico Institute of Mining and Technology, Socorro, NM, and the Naval Air Warfare Center Weapons Division, China Lake, CA. The code has general applicability, but has been most exercised toward simulation of shock initiation and sympathetic detonation of explosives.

SMERF is a combination of procedures used in many well-known and established hydrocodes. The Lagrangian phase of the calculation uses a modified Lax-Wendroff method (Reference 1). The fluxing of materials in the Eulerian phase employs the van Leer's advection scheme (Reference 2) and the Simple Line Interface Calculation (SLIC) algorithm (Reference 3). These are standard techniques that have proven to be robust and accurate numerical procedures through years of use in the computational continuum dynamics community. The strength and uniqueness of **SMERF** lies in the detonation physics package. Several high-explosive burn models are available in the code, and they enable the user to simulate a wide range of complex problems involving energetic materials. Sympathetic detonation and shock initiation are examples of problems in which **SMERF** excels.

In Section II of this manual, the differential equations of motion, along with their finite difference approximations, are presented, and the solution procedure discussed. Section III describes inputs and outputs. Two example problems (shock initiation of high explosive by fragment impact, and sympathetic detonation in a bomb stack) useful in learning how to run **SMERF** are discussed in detail in Section IV.

II. UNDERSTANDING SMERF

A. GOVERNING EQUATIONS

The equations of fluid motion are written in axis-symmetric coordinates below. An elastic constitutive model and general functional forms for energetic reactions and equations of state are also included.

Conservation Equations:

Mass

$$\frac{\partial \rho}{\partial t} + U\frac{\partial \rho}{\partial r} + V\frac{\partial \rho}{\partial z} = -\rho\left[\frac{1}{r}\frac{\partial rU}{\partial r} + \frac{\partial V}{\partial z}\right] \qquad \text{A1}$$

Radial Momentum

$$\frac{\partial U}{\partial t} + U\frac{\partial U}{\partial r} + V\frac{\partial U}{\partial z} = -\frac{1}{\rho}\left[\frac{\partial (P - S_{rr})}{\partial r} + \frac{\partial S_{zz}}{\partial z} + \frac{2S_{rr} + S_{zz}}{r}\right] \qquad \text{A2}$$

Axial Momentum

$$\frac{\partial V}{\partial t} + U\frac{\partial V}{\partial r} + V\frac{\partial V}{\partial z} = -\frac{1}{\rho}\left[\frac{\partial (P - S_{zz})}{\partial r} + \frac{\partial S_{rz}}{\partial r} + \frac{S_{rz}}{r}\right] \qquad \text{A3}$$

Total Energy

$$\frac{\partial E}{\partial t} + U\frac{\partial E}{\partial r} + V\frac{\partial E}{\partial z} = -\frac{1}{\rho}\left[\frac{1}{r}\frac{\partial rUP}{\partial r} + \frac{\partial VP}{\partial z}\right] + \qquad \text{A4}$$

$$\frac{1}{\rho}\left[\left(\frac{\partial US_{rr}}{\partial r} - \frac{US_{rr}}{r}\right) + \left(\frac{\partial VS_{zz}}{\partial z} - \frac{US_{zz}}{r}\right) + \left(\frac{\partial US_{rz}}{\partial z} + \frac{\partial VS_{rz}}{\partial r}\right)\right]$$

Internal Energy

$$I = E - \tfrac{1}{2}\left(U^2 + V^2\right) \qquad \text{A5}$$

Constitutive Equations:

$$\frac{\partial S_{rr}}{\partial t} + U\frac{\partial S_{rr}}{\partial r} + V\frac{\partial S_{rr}}{\partial z} = 2\mu\left[\frac{\partial U}{\partial r} - \frac{1}{3}D\right] + 2R'S_{rz} \qquad \text{A6}$$

$$\frac{\partial S_{zz}}{\partial t} + U\frac{\partial S_{zz}}{\partial r} + V\frac{\partial S_{zz}}{\partial z} = 2\mu\left[\frac{\partial V}{\partial z} - \frac{1}{3}D\right] - 2R'S_{rz} \qquad \text{A7}$$

$$\frac{\partial S_{rz}}{\partial t} + U\frac{\partial S_{rz}}{\partial r} + V\frac{\partial S_{rz}}{\partial z} = \mu\left[\frac{\partial U}{\partial z} + \frac{\partial V}{\partial r}\right] - R'\left(S_{rr} - S_{zz}\right) \qquad \text{A8}$$

Equation of State:

$$P = P(\rho, I, W) \qquad \text{A9}$$

Energetic Reaction Rate:

$$\frac{\partial W}{\partial t} + U\frac{\partial W}{\partial r} + V\frac{\partial W}{\partial z} = f(P, T, W) \qquad \text{A10}$$

Quantities appearing in the above equations are defined in Table 1.

TABLE 1. Definition of Variables and Units.

Variable	Definition	Variable	Definition
r	radial coordinate (cm)	P	pressure (Mb)
z	axial coordinate (cm)	S	deviatoric stress (Mb)
t	time coordinate (μs)	D	divergence (1/μs)
ρ	density (g/cm^3)	R'	spin tensor (1/μs)
I	internal energy (Mb-cm^3/g)	μ	shear modulus (Mb)
E	total energy (Mb-cm^3/g)	W	burn fraction or porosity
U	radial velocity (cm/μs)	T	temperature (K)
V	axial velocity (cm/μs)		

The equations are written in a reference frame fixed with respect to the laboratory (i.e., an Eulerian frame). In this frame, a fluid element moves through space with the local fluid velocity and changes because of velocity gradients. The second term on the left-hand-side of the time-dependent equations is due to the motion of the fluid with respect to the laboratory frame and is called the advection term. When solving the equations, it is convenient to treat these terms separately. First, the equations without the advection term (Lagrangian equations) are solved. The result is what an observer traveling along with a fluid element would see and, therefore, gives the solution in this moving frame. To get the desired result, the Lagrangian solution is mapped back to the fixed Eulerian frame. This mapping is called fluxing (as well as advection) because one can easily picture it in terms of fluid moving through surfaces. After the differential equations are solved, the fluid pressure can be obtained. It is natural, therefore, to divide the solution procedure into three parts: 1) Lagrangian phase for source terms, 2) Eulerian phase for the remap or fluxing, and 3) pressure calculation in the equation of state.

B. NUMERICAL SOLUTION

1. LAGRANGIAN PHASE

The first step in the solution procedure is to solve the time dependent equations without the advection term. This step is called the Lagrangian step, as it provides the solution in the moving Lagrangian frame. A tilde symbol (~) is used to denote the Lagrangian updated values.

Conservation Equations:

$$\tilde{\rho}_{i,j}^{n+1} = \rho_{i,j}^{n}\left(1 - \delta t D_{i,j}^{n+1/2}\right) \qquad \text{B1.1}$$

$$\tilde{U}_{i,j}^{n+1} = U_{i,j}^{n} - \frac{\delta t}{\rho_{i,j}^{n}} \frac{P_{i+1/2,j}^{n+1/2} - P_{i-1/2,j}^{n+1/2}}{\delta r} \qquad \text{B1.2}$$

$$\tilde{V}_{i,j}^{n+1} = V_{i,j}^{n} - \frac{\delta t}{\rho_{i,j}^{n}} \frac{P_{i,j+½}^{n+½} - P_{i,j-½}^{n+½}}{\delta z} \qquad \text{B1.3}$$

$$\tilde{E}_{i,j}^{n+1} = E_{i,j}^{n} - \frac{\delta t}{\rho_{i,j}^{n}} \left[\frac{(rUP)_{i+½,j}^{n+½} - (rUP)_{i-½,j}^{n+½}}{r_i \delta r} + \frac{(VP)_{i,j+½}^{n+½} - (VP)_{i,j-½}^{n+½}}{\delta z} \right] \qquad \text{B1.4}$$

Constitutive Equations:

$$\left(\tilde{S}_{rr}\right)_{i,j}^{n+1} = (S_{rr})_{i,j}^{n} + 2\mu\delta t \left(\frac{U_{i+½,j}^{n+½} - U_{i-½,j}^{n+½}}{\delta r} - \frac{1}{3} D_{i,j}^{n+½} \right) + 2R'(S_{rz})_{i,j}^{n} \delta t \qquad \text{B1.5}$$

$$\left(\tilde{S}_{zz}\right)_{i,j}^{n+1} = (S_{zz})_{i,j}^{n} + 2\mu\delta t \left(\frac{V_{i,j+½}^{n+½} - V_{i,j-½}^{n+½}}{\delta z} - \frac{1}{3} D_{i,j}^{n+½} \right) - 2R'(S_{rz})_{i,j}^{n} \delta t \qquad \text{B1.6}$$

$$\left(\tilde{S}_{rz}\right)_{i,j}^{n+1} = (S_{rz})_{i,j}^{n} + \mu\delta t \left(\frac{U_{i,j+½}^{n+½} - U_{i,j-½}^{n+½}}{\delta z} + \frac{V_{i+½,j}^{n+½} - V_{i-½,j}^{n+½}}{\delta r} \right) \qquad \text{B1.7}$$

$$- R' \left[(S_{rr})_{i,j}^{n} - (S_{zz})_{i,j}^{n} \right] \delta t$$

Equation of State:

$$P_{i,j}^{n+1} = f(\rho_{i,j}^{n+1}, I_{i,j}^{n+1}, W_{i,j}^{n+1}) \qquad \text{B1.8}$$

Energetic Reactions:

$$W_{i,j}^{n+1} = f(W_{i,j}^{n}, P_{i,j}^{n}) \qquad \text{B1.9}$$

Divergence:

$$D_{i,j}^{n+1} = \frac{r_{i+½} U_{i+½,j}^{n+½} - r_{i-½} U_{i-½,j}^{n+½}}{r_i \delta r} + \frac{V_{i,j+½}^{n+½} - V_{i,j-½}^{n+½}}{\delta z} \qquad \text{B1.10}$$

Material strength terms have been omitted from the conservation equations above. These contributions are given separately below.

$$\tilde{U}_{i,j}^{n+1} = \tilde{U}_{i,j}^{n+1} + \frac{\delta t}{\rho_{i,j}^{n}} \left[\frac{(S_{rr})_{i+½,j}^{n+½} - (S_{rr})_{i-½,j}^{n+½}}{\delta r} + \frac{(S_{rz})_{i,j+½}^{n+½} - (S_{rz})_{i,j-½}^{n+½}}{\delta z} + \frac{2(S_{rr})_{i,j}^{n+½} + (S_{zz})_{i,j}^{n+½}}{r_i} \right] \qquad \text{B1.11}$$

$$\tilde{V}_{i,j}^{n+1} = \tilde{V}_{i,j}^{n+1} + \frac{\delta t}{\rho_{i,j}^{n}} \left[\frac{(S_{zz})_{i,j+½}^{n+½} - (S_{zz})_{i,j-½}^{n+½}}{\delta z} + \frac{(S_{rz})_{i+½,j}^{n+½} - (S_{rz})_{i-½,j}^{n+½}}{\delta r} + \frac{(S_{rz})_{i,j}^{n+½}}{r_i} \right] \qquad \text{B1.12}$$

$$\tilde{E}_{i,j}^{n+1} = \tilde{E}_{i,j}^{n+1} + \frac{\delta t}{\rho_{i,j}^{n}} \left[\frac{(US_{rr})_{i+½,j}^{n+½} - (US_{rr})_{i-½,j}^{n+½}}{\delta r} + \frac{(VS_{zz})_{i,j+½}^{n+½} - (VS_{zz})_{i,j-½}^{n+½}}{\delta z} + \frac{(US_{rr})_{i,j}^{n+½} + (VS_{rz})_{i,j}^{n+½}}{r_i} \right. \qquad \text{B1.13}$$

$$\left. + \frac{(VS_{rz})_{i+½,j}^{n+½} - (VS_{rz})_{i-½,j}^{n+½}}{\delta r} + \frac{(US_{rz})_{i,j+½}^{n+½} - (US_{rz})_{i,j-½}^{n+½}}{\delta z} \right]$$

In the differenced equations, δr and δz are the horizontal and vertical dimensions of a grid cell and δt is the time step. Subscripts i and j indicate the space centering and superscript n the time centering. Quantities at cell boundaries and time (n) are weighted averages. Quantities at the half-time step ($n+1/2$) and cell boundaries ($i\pm1/2$) and ($j\pm1/2$) are computed as follows:

$$U^{n+½}_{i+½,j} = U^{n}_{i+½,j} - \frac{\delta t/2}{\rho^{n}_{i+½,j}}\left[\frac{P^{n}_{i+1,j} - P^{n}_{i,j}}{\delta r}\right] \qquad \text{B1.14}$$

$$V^{n+½}_{i,j+½} = V^{n}_{i,j+½} - \frac{\delta t/2}{\rho^{n}_{i,j+½}}\left[\frac{P^{n}_{i,j+1} - P^{n}_{i,j}}{\delta z}\right] \qquad \text{B1.15}$$

$$P^{n+½}_{i+½,j} = P^{n}_{i+½,j} - \frac{\delta t}{2}\left(\rho c^2\right)^{n}_{i+½,j}\left[\frac{(rU)^{n}_{i+1,j} - (rU)^{n}_{i,j}}{r_{i+½}\delta r} + \frac{V^{n}_{i+½,j+½} - V^{n}_{i+½,j-½}}{\delta z}\right] \qquad \text{B1.16}$$

$$P^{n+½}_{i,j+½} = P^{n}_{i,j+½} - \frac{\delta t}{2}\left(\rho c^2\right)^{n}_{i,j+½}\left[\frac{(rU)^{n}_{i+½,j+½} - (rU)^{n}_{i-½,j+½}}{r_i\delta r} + \frac{V^{n}_{i,j+1} - V^{n}_{i,j}}{\delta z}\right] \qquad \text{B1.17}$$

The half-step pressure update has used $dP/dt=-\rho c^2 D$, a prognostic equation which can be derived using Equations A1-A3. The half-step stresses are computed as follows.

$$(S_{rr})^{n+½}_{i+½,j} = (S_{rr})^{n}_{i+½,j} - 2\mu\delta t\left[\frac{U^{n}_{i+1,j} - U^{n}_{i,j}}{\delta r} - \frac{1}{3}D^{n}_{i+½,j}\right] \qquad \text{B1.18}$$

$$(S_{zz})^{n+½}_{i,j+½} = (S_{zz})^{n}_{i,j+½} - 2\mu\delta t\left[\frac{V^{n}_{i,j} - V^{n}_{i,j+1}}{\delta r} - \frac{1}{3}D^{n}_{i,j+½}\right] \qquad \text{B1.19}$$

$$(S_{rz})^{n+½}_{i+½,j} = (S_{rz})^{n}_{i+½,j} - \mu\delta t\left[\frac{U^{n}_{i+1,j} - U^{n}_{i,j}}{\delta r} + \frac{V^{n}_{i+½,j+½} - V^{n}_{i+½,j-½}}{\delta z} - \frac{1}{3}D^{n}_{i+1/2,j}\right] \qquad \text{B1.20}$$

$$(S_{rz})^{n+½}_{i,j+½} = (S_{rz})^{n}_{i,j+½} - \mu\delta t\left[\frac{U^{n}_{i+½,j+½} - U^{n}_{i-½,j+½}}{\delta r} + \frac{V^{n}_{i,j+1} - V^{n}_{i,j}}{\delta z} - \frac{1}{3}D^{n}_{i,j+½}\right] \qquad \text{B1.21}$$

2. EULERIAN PHASE

The second step in the solution procedure is to map the solution obtained in step 1 (Lagrangian update) back onto the fixed Eulerian mesh. This is accomplished by moving material across zone boundaries. A one-dimensional algorithm is used for the flux calculations, with each direction treated separately. Consider the movement of mass. Let $\delta\tau$ be the flux volume swept out by moving the grid line at the right edge of cell (*id,j*) back to the fixed Eulerian position. We use an upstream scheme in which the donor cell value of ρ is evaluated at the center of the flux volume using one of three candidate slopes (*Sl, Sc, Sr*). These concepts are illustrated in Figure 1.

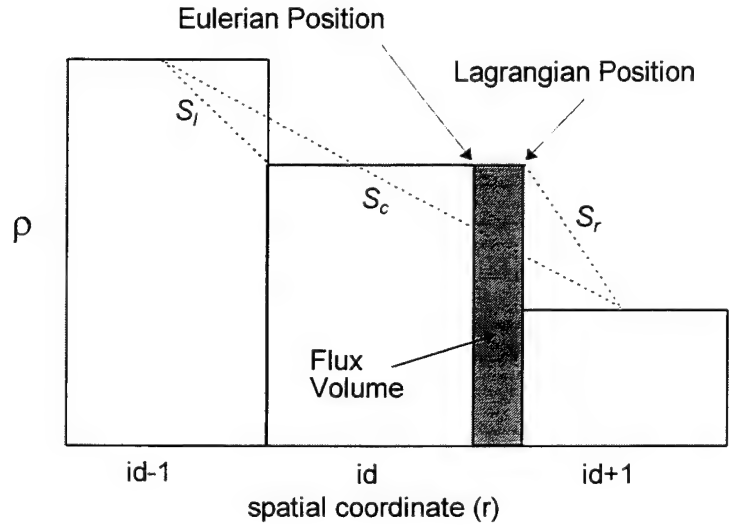


FIGURE 1. Illustration of van Leer's Advection Scheme.

The flux volume is

$$\delta\tau = \pi\left[\left(r_{i+\frac{1}{2}} + \tilde{U}^{n+1}_{i+\frac{1}{2},j}\delta t\right)^2 - r^2_{i+\frac{1}{2}}\right]\delta z \qquad \text{B2.1}$$

The mass in $\delta\tau$ crossing the *i+1/2* boundary is

$$\delta m_{i+\frac{1}{2},j} = \left[\rho_{id,j} + \frac{1}{2}\frac{d\rho}{dr}\left(\delta r - \tilde{U}^{n+1}_{i+\frac{1}{2},j}\delta t\right)\right]\delta\tau = \rho^*\delta\tau \qquad \text{B2.2}$$

where the density ρ^* is evaluated at the center of the flux volume using the smallest of three candidate slopes

$$S_l = \frac{\rho_{id,j} - \rho_{id-1,j}}{\delta r/2}\ ,\ S_c = \frac{\rho_{id+1,j} - \rho_{id-1,j}}{2\delta r}\ ,\ S_r = \frac{\rho_{id+1,j} - \rho_{id,j}}{\delta r/2} \qquad \text{B2.3}$$

$$\frac{d\rho}{dr} = \begin{cases} MIN(|S_l|,|S_c|,|S_r|) * SIGN(S_l * S_r) & \text{if } S_l * S_r > 0 \\ 0 & \text{if } S_l * S_r < 0 \end{cases} \qquad \text{B2.4}$$

The new (Eulerian updated) mass in the cell is simply the net flux,

$$m_{i,j}^{n+1} = m_{i,j}^{n} + \delta m_{i+\frac{1}{2},j} - \delta m_{i-\frac{1}{2},j} \qquad \text{B2.5}$$

The fluxing of other quantities is handled in the same way as the mass. Letting Ψ represent the momentum or total energy and ψ the mass specific values (velocity or specific total energy), the amount of Ψ crossing a cell boundary is $\delta\Psi=\psi\delta m$. Where ψ, like ρ, is evaluated at the center of the flux volume. Conservation of Ψ then gives the new (Eulerian updated) values of ψ.

$$\psi_{i,j}^{n+1} = \frac{m_{i,j}^{n}\psi_{i+\frac{1}{2},j} + \delta\Psi_{i+\frac{1}{2},j} - \delta\Psi_{i-\frac{1}{2},j}}{m_{i,j}^{n+1}} \qquad \text{B2.6}$$

The fluxing scheme described above is van Leer's modification (Reference 2) of the upstream advection scheme. The algorithm is second-order accurate and does not introduce spurious oscillations common to higher order schemes. When more than one material is present in the donor cell, its kind and the order in which the flux volume is filled must be determined. The SLIC (Simple-Line-Interface-Calculation) algorithm (Reference 3) is used for this purpose. SLIC looks in the zone ahead (downstream) and behind (upstream) the donor cell and assigns one of four index number pairs to each material in the donor cell. The pair (0,0) means that none of this particular material is present in the zone behind the donor cell and none is present in the zone ahead of the donor cell. Accordingly, (0,1) means that there is none behind and some ahead, (1,0) indicates some behind and none ahead, and (1,1) says that there is some of this material in the zones both upstream and downstream of the donor cell. For positive velocities, the materials are fluxed in the order [(0,1),(1,1),(0,0),(1,0)] and, for negative velocities, the order is [(1,0),(1,1),(0,0),(0,1)]. Figure 2 depicts the algorithm for a simple case in which material flows from the center cell containing two different materials into the cell on the right. A two-dimensional example of how SLIC represents a fluid interface is shown in Figure 3.

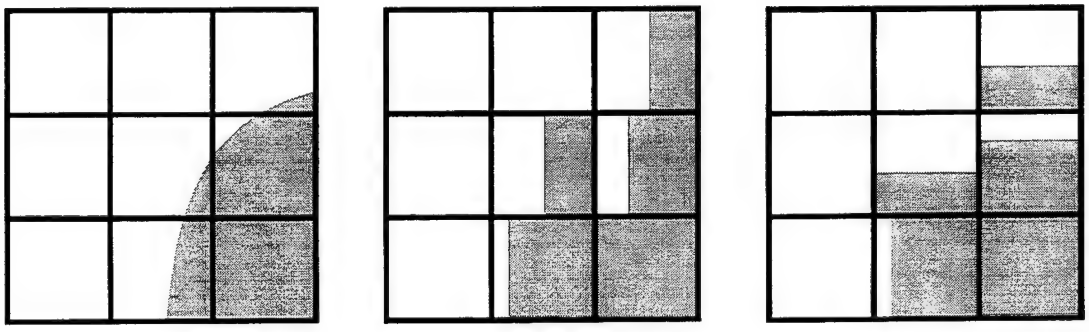


FIGURE 2. Illustration of How the SLIC Algorithm Represents an Interface for Movement in Two Directions.

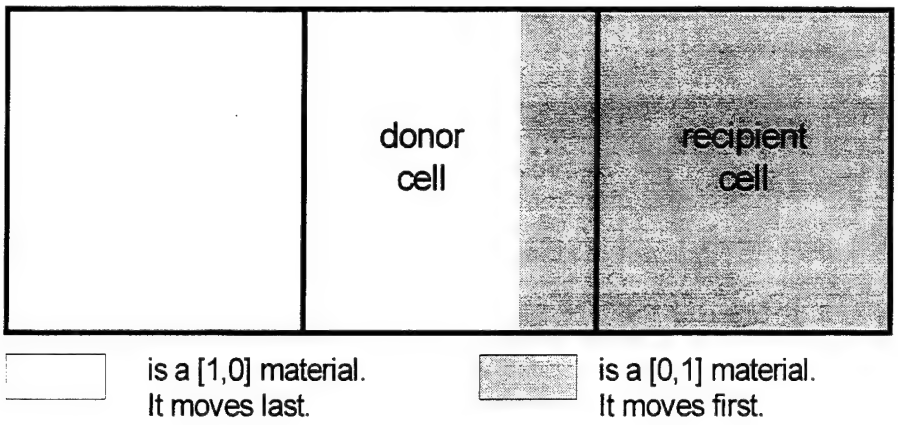


FIGURE 3. Simple Illustration of How the SLIC Algorithm Assigns an Order to the Movement of Material Across a Cell Boundary.

C. EQUATIONS OF STATE

1. ANALYTIC FORMS

Step 3 in the solution procedure consist in calculating the pressure from the newly determined density and internal energy. Four equations of state are available: Perfect Gas, Gruneisen, HOM (Reference 4), and JWL (Reference 5). In the following equations, Hugoniot and isentrope reference curves are indicated by subscripts *"h"* and *"i,"* respectively. The compression is $\mu=\rho/\rho_o-1$, and e_O is the heat of reaction normalized by the initial specific volume.

Perfect Gas:

$$P=(\gamma-1)\rho I \qquad \text{C1.2}$$

Gruneisen:

$$P=\frac{\rho_o C^2\mu\left[1+(1-\gamma/2)\mu-a\mu^2/2\right]}{\left[1-(S_1-1)\mu-\frac{S_2\mu^2}{(\mu+1)}-\frac{S_3\mu^3}{(\mu+1)^3}\right]^2}+(\gamma+a\mu)I \quad \mu>1 \qquad \text{C1.3a}$$

$$P=\rho_o c\mu+(\gamma_o+a\mu)I \quad \mu<1 \qquad \text{C1.3b}$$

HOM Solid:

$$P=P_h+\frac{\gamma}{V}(I-I_h) \quad V_o/V>1 \qquad \text{C1.4a}$$

$$P = \left[I - \frac{C_v}{3\alpha}\left(\frac{V}{V_o} - 1\right)\right]\frac{\gamma}{V} \qquad V_o/V < 1 \qquad \text{C1.4b}$$

$$P_h = \frac{C^2(V_o - V)}{\left[V_o - S(V_o - V)\right]^2} \qquad \text{C1.4c}$$

$$I_h = \frac{1}{2} P_h (V_o - V) \qquad \text{C1.4d}$$

$$T = T_h + (I - I_h)/C_v \qquad \text{C1.4e}$$

$$\ln T_h = F + G(\ln V) + H(\ln V)^2 + I(\ln V)^3 + J(\ln V)^4 \qquad \text{C1.4f}$$

HOM Gas:

$$P = P_i + \frac{1}{\beta}(I - I_i)/V \qquad \text{C1.5a}$$

$$T = T_i + (I - I_i)/C_v \qquad \text{C1.5b}$$

$$\ln P_i = A + B(\ln V) + C(\ln V)^2 + D(\ln V)^3 + E(\ln V)^4 \qquad \text{C1.5c}$$

$$\ln T_i = Q + R(\ln V) + S(\ln V)^2 + T(\ln V)^3 + U(\ln V)^4 \qquad \text{C1.5d}$$

$$\ln(I_i + Z) = K + L(\ln P) + M(\ln P)^2 + N(\ln P)^3 + O(\ln P)^4 \qquad \text{C1.5e}$$

$$-1/\beta = R + 2S(\ln V) + 3T(\ln V)^2 + 4U(\ln V)^3 \qquad \text{C1.5f}$$

JWL:

$$P = A\left(1 - \frac{wV_o}{R_1 V}\right)e^{-R_1 V/V_o} + B\left(1 - \frac{wV_o}{R_2 V}\right)e^{-R_2 V/V_o} + w\left(\frac{I}{V_o} + e_o\right) \qquad \text{C1.6a}$$

$$T = \left(\frac{A}{R_1}e^{-R_1 V} + \frac{B}{R_2}e^{-R_2 V} + \frac{I}{V_o} + e_o\right)\frac{1}{C_v} \qquad \text{C1.6b}$$

2. ADIABATIC PRESSURE EQUILIBRIUM IN MIXED CELLS

Some procedure for calculating the pressure in computational cells containing different kinds of materials is needed because the partial pressures are generally not equal after the Lagrangian and Eulerian steps. We use a Newton-Raphson iterative procedure, adjusting the partial volumes (τ_i) along adiabats, until the partial pressures (P_i) are equal. Consider the simple case of two materials. The mass ($M=M_1+M_2$), volume ($\tau=\tau_1+\tau_2$), and internal energy ($I=M_1I_1+M_2I_2$) of the cell as a whole are known and serve as constraints during the iteration. Requiring the partial pressures equal to the first order on the k+1 iteration step gives

$$P_1^{k+1} = P_1^k + \frac{dP_1}{dV_1}\frac{d\tau_1}{M_1} = P_2^{k+1} = P_2^k + \frac{dP_2}{dV_2}\frac{d\tau_2}{M_2} \qquad \text{C2.1}$$

where the derivatives are taken along adiabats

$$\frac{dP}{dV} = \frac{\partial P}{\partial V} + \frac{\partial P}{\partial I}\frac{dI}{dV} = \frac{\partial P}{\partial V} - P\frac{\partial P}{\partial I} \qquad \text{C2.2}$$

Using the volume constraint ($d\tau_2 = -d\tau_1$), we solve for the change in partial volumes.

$$d\tau_1 = (P_2^k - P_1^k)\left(\frac{1}{M_1}\frac{dP_1}{dV_1} + \frac{1}{M_2}\frac{dP_2}{dV_2}\right)^{-1} \qquad \text{C2.3}$$

The accompanying energy changes along the adiabats are $M_1 dI_1 = -P_1 d\tau_1$ and $M_2 dI_2 = -P_2 d\tau_2$. Since this procedure does not conserve the total energy in the cell, the cell pressure $P^{k+1} = P_1^{k+1} = P_2^{k+1}$ defined by C2.1 is used as follows

$$I_1^{k+1} = I_1^k - P^{k+\frac{1}{2}}\frac{d\tau_1}{M_1} \qquad \text{C2.4}$$

$$I_2^{k+1} = I_2^k - P^{k+\frac{1}{2}}\frac{d\tau_2}{M_2} \qquad \text{C2.5}$$

which conserves the energy during each step of the iteration. The pressure computation is done after the Lagrangian step and after the Eulerian step. On entering the mixed cell routine, the sum of the initial material volumes and energies does not equal the cell total volume and energy. Therefore, prior to the pressure equilibration calculation, the excess volume is distributed to the materials on a volume basis, and the excess energy is distributes on a mass basis.

3. THERMODYNAMIC EQUILIBRIUM FOR EXPLOSIVES

Some procedure is required for calculating the pressure in computational cells containing a mixture of reactants and products. The method we use is to bring the constituents to pressure and temperature equilibrium by adjusting the partial volumes and energies, while conserving the total volume and energy in the cell. Let us consider the case where the mixed cell contains unreacted solid (*s*) and reaction product gas (*g*). We have the solid and gas equations of state

$$P_s = P_s(V_s, I_s) \qquad \text{C3.1}$$

$$P_g = P_g(V_g, I_g) \qquad \text{C3.2}$$

The specific volume and internal energy are partitioned into unreacted solid and gaseous products by the conservation equations and the unburned fraction *W*.

$$V = WV_s + (1-W)V_g \qquad \text{C3.3}$$

$$I = WI_s + (1-W)I_g \qquad \text{C3.4}$$

We assume the solid and gas phases are in pressure and temperature equilibrium.

$$P_s = P_g \tag{C3.5}$$

$$T_s = T_g \tag{C3.6}$$

Introduction of the temperature necessitates a thermal equation of state, which we take to be linear in the temperature

$$I_s = I_{os}(V_s) + C_{vs}(V_s)T_s \tag{C3.7}$$

$$I_g = I_{og}(V_g) + C_{vg}(V_g)T_g \tag{C3.8}$$

Substituting C3.7 and C3.8 into C3.4 and imposing C3.6 yields

$$I = I_o + C_v T \tag{C3.9}$$

where

$$I_o = WI_{os}(V_s) + (1-W)I_{og}(V_g) \tag{C3.10}$$

$$C_v = WC_{vs}(V_s) + (1-W)C_{vg}(V_g) \tag{C3.11}$$

An equation for the temperature is obtained from C3.9 and C3.3

$$T = T(V, I, W, V_g) \tag{C3.12}$$

Every quantity is known, except for V_g. Once the equilibrium temperature is known, equations C3.7 and C3.8 give the partial energies, I_g and I_s, and then, finally, C3.1 and C3.2 give the partial pressures P_g and P_s . One gets a relation for the pressure difference dP as a function of the unknown gas volume V_g and there is a particular choice of V_g for which the pressure equilibrium ($dP=0$) holds. Newton's method is used to solve for V

D. HIGH EXPLOSIVE BURN MODELS

An equation describing the rate of decomposition of energetic materials is required for the simulation of detonation phenomena. Four burn models are available: Programmed, Arrhenius, Nucleation & Growth, and Forest Fire.

1. PROGRAMMED BURN

This model is used to generate a detonation wave in an explosive which travels at the Chapman-Jouget velocity (D_{CJ}). The detonation is started at a particular location which is specified in the input. The detonation is assumed to start at time zero. At the proper time, the explosive reactants within a cell are converted to products.

2. ARRHENIUS BURN

The Arrhenius rate

$$\frac{dW}{dt} = WZ^* e^{-E^*/R_g T} \qquad \text{D2.1}$$

involves two parameters: the activation energy (E^*), and a frequency factor (Z^*).

3. NUCLEATION & GROWTH

The Nucleation and Growth burn has the most complex form of all burn models in the code. The burn rate consists of three terms: an ignition term and two growth terms.

$$\frac{dW}{dt} = \frac{dW}{dt}\Big)_{ig} + \frac{dW}{dt}\Big)_{g1} + \frac{dW}{dt}\Big)_{g2} \qquad \text{D3.1}$$

Each term can be switched on or off depending on the product mass fraction. The ignition rate requires that the compression of the material be greater than a threshold value and that the rate vanish after some fraction of the material is burned. The individual rates are

$$\frac{dW}{dt}\Big)_{ig} = \begin{cases} G_o\left[\left(V_o/V - 1\right) - C_{cr}\right]^{n1} & \left(1-W\right) > F_{mxig} \\ 0 & \left(1-W\right) < F_{mxig} \end{cases} \qquad \text{D3.2}$$

$$\frac{dW}{dt}\Big)_{g1} = \begin{cases} G_1 W^{\delta_1}\left(1-W\right)^{r_1} P^m & \left(1-W\right) > F_{mxgr} \\ 0 & \left(1-W\right) < F_{mxgr} \end{cases} \qquad \text{D3.3}$$

$$\frac{dW}{dt}\Big)_{g2} = \begin{cases} G_2 W^{\delta_2}\left(1-W\right)^{r_2} P^n & \left(1-W\right) > F_{mngr} \\ 0 & \left(1-W\right) < F_{mngr} \end{cases} \qquad \text{D3.4}$$

The 14 parameters in the Nucleation and Growth model are determined by comparing experimental data with theoretical calculations. The best experiments for this purpose use embedded gauges which record the response of the explosive to planar shocks. Model parameters are then adjusted to obtain agreement between gauge records and calculations.

The Nucleation and Growth burn model originated with Lee and Tarver (Reference 6) and was expanded by Tarver and Hallquist (Reference 7) to its present form. In general, the values for the nucleation and growth constants depend on the choice of equation of state for the reactants and products and the rule governing their mixtures. In previous work, the JWL equation of state was used for both phases. However, the mixture rule was used, not the equilibrium temperature and pressure rule used in **SMERF**. Therefore, nucleation and growth constants found in the literature cannot be used in **SMERF.**

4. FOREST-FIRE

The Forest Fire burn model is a global reaction kinetics model for the decomposition of heterogeneous explosives. The burn model is calibrated directly using the results of wedge tests. Analytic methods for calibrating the model were derived by Forest (Reference 8), the originator of the model. Lundstrom (Reference 9) has reviewed and extended Forest's original work.

In a wedge test, a shock wave is introduced into the test explosive. Due to burning in the shocked explosive, the shock wave accelerates and transitions into a detonation wave. The progress of the accelerating shock wave is monitored experimentally using the wedge shape of the explosive. The distance that the shock wave takes to transition to detonation is called the run distance. The experimental data is customarily displayed as a log-log plot of the run distance (X^*) as a function of the initial shock pressure (P). This representation of the data is known as a Pop Plot, after its originator A. Popolato (Reference 10). For the vast majority of explosives, data on the Pop Plot can be fit very well with a straight line, in which case the wedge test results can be expressed in the form

$$X^* = \left(\frac{P_1}{P}\right)^S \qquad \text{D4.1}$$

where S is the slope of the line and P_1 is the pressure (Kb) required to give a one-centimeter run distance. The result of each wedge test is a shock trajectory from which the shock pressure as a function of the shock propagation distance and time can be obtained. An important assumption in Forest's derivation is the "single curve buildup principle". According to this principle, all of the wedge test shock trajectories corresponding to different initial shock pressures can be reduced to a single curve. The trajectories must first be shifted in time and space to superimpose their points of transition to detonation. Equation D4.1, therefore, describes the resulting single trajectory.

In Forest's method, as described by Lundstrom (Reference 9), one calculates the burn rate behind the shock wave which is accelerating according to equation D4.1. To do this, one uses the differential equations expressing the conservation of mass, momentum, and energy in the reacting flow field and the Hugoniot equations expressing the same conservation principles across the shock wave. An expression can be derived for the burn rate behind the shock wave having the functional form

$$\frac{dW}{dt} = F(P,W)\frac{dP}{dX^*} + G(P,W)\frac{\partial P}{\partial x} \qquad \text{D4.2}$$

where dP/dX^* is the change of shock pressure with run distance as given by equation D4.1, and partial derivative is the pressure gradient immediately behind the shock front. The expressions F and G are known functions of the shock pressure (P) and the reactant mass fraction (W) behind the shock. In general, one can assume explosive decomposition within the shock wave so that W can be less than one. One normally makes the simplifying assumption that $\partial P/\partial X=0$, since no information regarding this quantity is obtained from the wedge test. Together with equation D4.1 for dP/dX^*, equation D4.2 reduces to

$$\frac{dW}{dt} = W(P,W) \qquad \text{D4.3}$$

That is, the burn rate required to accelerate the shock wave is a known function of shock pressure and degree of reaction within the shock wave. This function can then be used to calibrate different burn models. Forest assumed that the reaction of the explosive takes place according to a first-order decomposition reaction. First-order reactions rates have the form

$$\frac{1}{W}\frac{dW}{dt} = f(P) \qquad \text{D4.4}$$

where the reaction rate function, *f(P)*, is independent of the solid mass fraction and is expressed solely as a function of the local pressure.

An alternative to the first-order hypothesis is the ignition and growth concept used by Lee and Tarver to describe explosive burn. The shock initiation of the heterogeneous solid explosive is modeled by ignition at localized hot spots followed by hole burning at the growing hot spot boundaries. This model has had some success in correlating the detailed mass, velocity, and pressure histories behind the shock wave during shock-to-detonation experiments, lending credibility to the basic concept. According to the ignition and growth model, the global reaction rate is a minimum at the shock boundary where *W=1* because of the small burn surface area of the hot spots. This is in direct opposition to the assumption of first-order kinetics, where the reaction is maximum at the shock front.

Lundstrom (Reference 9) presented an alternative zero-order kinetics assumption where the explosive decomposition is given by the form

$$\frac{dW}{dt} = f(P) \qquad \text{D4.5}$$

which ignores all dependence of the reaction rate on the solid mass fraction. This model should lie somewhere between the extremes of the ignition and growth model and the first-order kinetics employed by Forest. Lundstrom discussed the calibration for both first and zero-order reaction rate kinetics using different assumptions on the degree of reaction within the shock wave. The resulting burn rate models were tested by performing numerical simulations of the wedge tests to see if the overall wedge test results, equation D4.1, are obtained. It was found that the zero-order reaction rate law gave reasonable results when calibrated using the assumption of no reaction with the shock wave. In particular, it was found that the reaction rate was well behaved at pressures beyond the Von Neumann spike pressure. The original Forest Fire model became infinitely near the Chapman-Jouget pressure. This is important when trying to apply the burn model to problems involving detonation failure. For an inert shock, the explicit form of equation D4.3 is

$$\frac{dW}{dt} = f(P) = -\left\{1+\left(\frac{PP_1 - P_V}{PP_1 - 2P_V}\right)\left[1-(V_o - V)\frac{P_V}{P}\right]\right\}\frac{U_s}{P_w}\frac{dP}{dX^*} \qquad \text{D4.6}$$

where

$$P_1 = \frac{dP}{dI} \qquad P_V = \frac{dP}{dV} \qquad P_W = \frac{dP}{dW} \qquad \text{D4.7}$$

and U_S is the shock velocity. Given an equation of state for the explosive reactants and products, the Hugoniot relations for the shock, and equation D4.1 for the run distance, the right hand side of D4.6 can be evaluated as a known function of the shock pressure.

The **SMERF** code uses the zero-order kinetics version of Forest Fire calibrated using the inert shock assumption. The function *f(P)* is evaluated at a number of points at the start of the **SMERF** calculation. They are used as a basis for interpolation using a cubic spline fit of *log(f)* as a function of *log(P)*.

E. POROUS MATERIALS

The porous material model in **SMERF** is based on the P-α model originally proposed by Herrmann (Reference 11). A similar model is incorporated into the CTH code in a substantially different form (Reference 12). A distention parameter, α, relates the overall specific volume of the porous material, V, to the specific volume of the solid matrix, V_s.

$$\alpha = \frac{V}{V_s} \qquad \text{E1}$$

Then the equation of state of the porous material is related to the solid matrix material by

$$P(V, I, \alpha) = \frac{1}{\alpha} P(V_s, I_s) \qquad \text{E2}$$

$$T(V, I, \alpha) = T(V_s, I_s) \qquad \text{E3}$$

A diagram of the equation of state of the porous material on the P-V plane is shown in Figure 4. The curves are shown for constant internal energy. The curve labeled $\alpha=1$ represents the compression curve for the solid matrix material. A curve labeled for constant $\alpha>1$ represents solutions of E2 for the distended porous material.

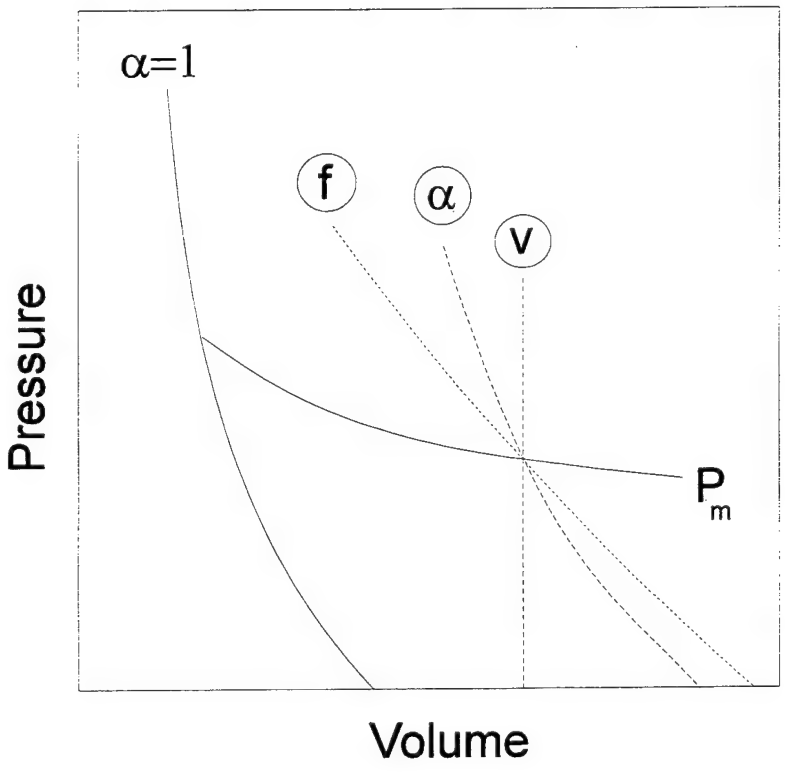


FIGURE 4. The Equation of State of a Porous Material in the P-V Plane With Constant Internal Energy.

Compression of the porous material is elastic and reversible at low pressure. The elastic compression curves can be described by equation E1 with α given as a function of pressure by

$$\alpha = f \cdot g(P,f) \qquad \text{E4}$$

where

$$g(0,f) = 1 \qquad \text{E5}$$

and

$$g(P,1) = 1 \qquad \text{E6}$$

According to E3 and E4, the parameter f is, therefore, the value of α at zero pressure. It is the extent of irreversible crushing that the porous material has undergone. The parameter, f, is a constant during the elastic compression of material. The shape of the elastic compression curve is given by the function $g(P,f)$ which will be specified later. Several elastic compression curves of constant f are included in Figure 4. The condition on the function g, Equation E5, ensures that the compression of the solid matrix material, α=1, is a special case of E3. This factoring of α into f and g is not the usual procedure, but it makes the geometry of the porous equation of state more physically intuitive (at least to the authors).

The porous material undergoes reversible, elastic compression until it reaches a limiting pressure designated by $P_m(V,I)$, where the compression becomes irreversible crushing. This crushing curve is included in Figure 4. The compression, therefore, first follows a constant f line until it intersects the P_m line, which it then follows until it intersects the solid matrix line α=1. Unloading follows the constant f line which intersects the P_m line at the maximum compression.

Experimentally, the solid matrix equation of state is often available or can be measured. The irreversible compaction curve, $P_m(V,I)$, can be measured in quasi-static compaction tests, and the data is frequently available in a tabular or a fitted functional form. Shock testing of the material can be performed to give the elastic shock precursor velocity that can be used to determine the $g(P,f)$ function. The precursor velocity is related to the slope of the constant f curve at zero pressure in Figure 4.

The constant f lines satisfy the implicit relation which results from combining E1, E2, and E4,

$$f \cdot g(P,f) \cdot P = P_s\left[\frac{V}{f \cdot g(P,f)}, I\right] \qquad \text{E6}$$

Partial derivatives of the pressure P with respect to V and E are readily calculated as

$$\alpha^2 \xi \frac{\partial P}{\partial V}\bigg)_I = \frac{\partial P_s}{\partial V_s}\bigg)_{I_s} \qquad \text{E7}$$

and

$$\alpha^2 \xi \frac{\partial P}{\partial I}\Big)_V = \frac{\partial P_s}{\partial I_s}\Big)_V \qquad \text{E8}$$

where the quantity ξ is given by

$$\xi = 1 + \left[P + \frac{V_s^2}{V} \frac{\partial P_s}{\partial V_s}\Big)_{I_s} \right] \frac{\partial \ln g}{\partial P} \qquad \text{E9}$$

The sound speed satisfies the thermodynamic identity

$$a^2 = -V^2 \left[\frac{\partial P}{\partial V}\Big)_I - P \frac{\partial P}{\partial I}\Big)_V \right] \qquad \text{E10}$$

so that, for the porous material, the sound speed is given by

$$a^2 = \frac{1}{\xi \alpha^2} a_s^2 \qquad \text{E11}$$

where a_s is the sound speed in the solid matrix material. The specific form for the function g is chosen to be

$$g(f, P) = \frac{1}{1 + hP(f-1)} \qquad \text{E12}$$

where h is a constant that is chosen to match the measured elastic sound speed, a_p at porosity f_p.

At $P=0$, then $g=1$, $\alpha=1$, and $a^2_s = -V^2_s \, dP_s/dV_s$, so that Equation E9 for ξ becomes

$$\xi(P=0) = 1 + h a_s^2 (f-1)/V \qquad \text{E13}$$

Substituting E13 for ξ in E11) and solving for h, one gets

$$h = \frac{V}{(f-1)} \left(\frac{1}{a^2} - \frac{1}{a_s^2} \right) \qquad \text{E14}$$

The calibration condition, sound speed $a=a_p$ at porosity, f_p determines h. The porous model is treated in **SMERF** as an equation of state with an internal degree of freedom, f. A subroutine is required that evaluates

$$P = P(V, I, f) \qquad \text{and} \qquad T = T(V, I, f) \qquad \text{E15}$$

To do this in principle, P is evaluated as if the material were on the elastic, reversible curve corresponding to the input compaction f. If the resulting pressure is greater than the crushing pressure, P_m, then a new compaction, $f = f_m$, is computed which satisfies

$$P_m(V, I) = P(V, I, f_m) \qquad \text{E16}$$

The subroutine then returns P, T, and the compaction, f, which may be modified to f_m if required. In practice, one first finds out which region the material is in (elastic, crushing, or fully crushed) and computes the crush pressure, $P_m = P_m(V,I)$. One can find out whether the material is fully crushed if the following inequality holds

$$P_m \le P_s(V,I) \tag{E17}$$

One sets $f=1$ and uses the equation of state for the fully crushed material. If not, then one estimates g as $g_m = g(P_m,f)$ and computes a trial pressure P_t,

$$P_t = \frac{1}{fg_m} P_s\left(\frac{V}{fg_m}, I\right) \tag{E18}$$

If

$$P_m < P_t \tag{E19}$$

then the material is crushing and one must solve

$$P_m = \frac{1}{f_m g(P_m,f)} P_s\left[\frac{V}{f_m g(P_m,f)}, I\right] \tag{E20}$$

for f_m. A non-linear second-order solver is used for this purpose. The solution, f_m, for the crushing material is known to be bounded by

$$f > f_m > 1 \tag{E21}$$

Finally, if the inequality E19 is not satisfied, then the material lies on the elastic curve and one must solve

$$P = \frac{1}{fg(P,f)} P_s\left[\frac{V}{fg(P,f)}, I\right] \tag{E22}$$

for the pressure P. The same non-linear solver is used in this step.

F. MATERIAL STRENGTH

An elastic, perfectly plastic, constitutive model is used to describe strength effects in solids. For linear elasticity (Reference 13) the rate of change of the deviatoric stresses is proportional to the traceless rate of strain.

$$\dot{S}_{\alpha\beta} = 2\mu\bar{\dot{\varepsilon}}_{\alpha\beta} = 2\mu\left(\dot{\varepsilon}_{\alpha\beta} - \dot{\varepsilon}_{\gamma\gamma}\delta_{\alpha\beta}\right) \tag{F1}$$

Here, μ is the shear modulus, and the strain rates are defined by

$$\dot{\varepsilon}_{\alpha\beta} = \left(\frac{\partial U_\alpha}{\partial x_\beta} + \frac{\partial U_\beta}{\partial x_\alpha}\right) \tag{F2}$$

The time derivative of the stress in Equation F1 is not objective; that is, the material response will change if the underlying frame with respect to which the stresses and deformations are measured undergoes a rigid translation or rotation. A variety of objective stresses have been formulated. The Jaumann rate is the most widely used in codes. Hermann (Reference 14) examines the relative merits of several stress rates. With the Jaumann rate, our constitutive relation is

$$\dot{S}_{\alpha\beta} - S_{\alpha\gamma}R_{\beta\gamma} - S_{\gamma\beta}R_{\alpha\gamma} = 2\mu\bar{\dot{\varepsilon}}_{\alpha\beta} = 2\mu\left(\dot{\varepsilon}_{\alpha\beta} - \dot{\varepsilon}_{\gamma\gamma}\delta_{\alpha\beta}\right) \qquad \text{F3}$$

where R is the rotation rate

$$R_{\alpha\beta} = \left(\frac{\partial U_\alpha}{\partial x_\beta} - \frac{\partial U_\beta}{\partial x_\alpha}\right) \qquad \text{F4}$$

The individual components of F3 are:

$$\frac{dS_{rr}}{dt} = 2\mu\left(\frac{\partial U}{\partial r} - \frac{1}{3}D\right) + 2R'S_{rz} \qquad \text{F5}$$

$$\frac{dS_{zz}}{dt} = 2\mu\left(\frac{\partial V}{\partial z} - \frac{1}{3}D\right) - 2R'S_{rz} \qquad \text{F6}$$

$$\frac{dS_{rz}}{dt} = \mu\left(\frac{\partial U}{\partial z} + \frac{\partial V}{\partial r}\right) - R'\left(S_{rr} - S_{zz}\right) \qquad \text{F7}$$

and in two-dimensional axis-symmetric coordinates,

$$R' = \frac{\partial U}{\partial z} - \frac{\partial V}{\partial r} \qquad \text{F8}$$

$$D = \frac{1}{r}\frac{\partial rU}{\partial r} + \frac{\partial V}{\partial z} \qquad \text{F9}$$

The plastic flow regime is determined by the von-Mieses criterion when the second stress invariant, $J^2 = S_{\alpha\beta}\,S_{\alpha\beta}$, exceeds the known flow stress, Y_o. The individual deviators are then brought back to the flow surface

$$S_{\alpha\beta} = S_{\alpha\beta}\sqrt{\frac{Y_o^2}{3J^2}} \qquad \text{F10}$$

III. RUNNING SMERF

A. BASIC INPUT (The *setup* File)

There are six input files which SMERF requires for problem initialization: **setup, mshgen, matcor, outdata, gaugein,** and an equation of state data file. The entries in all of the input files have the FORTRAN-free format where the data are separated by a comma or a space. The input files are prepared using any convenient text editor. Basic problem options are specified in file setup. One example problem discussed in Section IV is sympathetic detonation in a stack of four bombs. For this problem, the setup file looks like the one shown in Table 2.

The variable names appearing within the square brackets in Table 2 are for convenience and a memory aid and are ignored by the SMERF program. Entries in the **setup** file are described line-by-line below.

TABLE 2. Setup Input File for the Bomb Stack Example Problem.

'Bomb Stack'	[title]
0, 'p1', 0.0, 0.25	[niter , var , varmin , varmax]
F,'restartfile'	[restart , filnam]
1.0 , 1.0	[xscale , tscale]
200.0, 1200, 0.4	[tmax , maxcyl , safty]
F, .2, .3, 100,0,90.,90.	[lpexit,pexit,pratio,ncount,modestop,ntarget,ytmin,ytmax]
F, 0.0, 0.0, 0.0	[lslide , ntarget , ytmin , ytmax]
0	[igeom (0=Cartesian , 1=cylindrical)]
0, 1, 0, 1	[b1,b2,b3,b (0=r,1=t; l,r,b,t)]
T, 4, 0.0, 0.0	[lpburn , ipgburn , xdet , ydet]
T, 5	[lfburn , iffburn]
T	[lgauge]
T	[lstres]
T	[lpfail]
T, 0.0 , 5.0, '.'	[lplot , tplot, dtplot , splotdir]
2, '../eosdata'	[ieos , eosfilnam]
104, 104, 104, 20, 20, 1	[material EOS numbers]

Line #1. The problem title is input on the first line.

Line #2. The second line has four inputs having to do with running **SMERF** in an iteration mode. A common use of the **SMERF** code is to determine threshold conditions which lead to the detonation of an acceptor explosive material. The **niter** iterations will be done to converge on the threshold value of the parameter **var** which separates events that lead to a detonation from those that do not; **varmin** and **varmax** are bounds on **var**. If **niter** is zero, then no iterations will be performed and the remaining input on line 2 will be ignored (it should still be in the proper format, however). The parameter **var** is a two-character string (enclosed in single quotes) that signifies what parameter will be iterated. At present, there are four possibilities.:

1. **var** = 'p1': The shock sensitivity of an acceptor explosive material will be iterated. It is required that the Forest Fire burn model be used for the acceptor material. The input to this burn model is the straight line on the Pop plot as discussed in connection with equation D4.1. The quantity that will be iterated is P1, the pressure (in megabits here on **line #2**) required to give a one-centimeter run distance. The material number of the acceptor material is given by the parameter **iffburn** which is read in **line # 11**. Further input is needed to determine criteria for acceptor explosive detonation or no detonation. This input is obtained from **line #6**. If the problem stop time, **tmax**, or stop cycle, **maxcyl**, in **line # 5** are achieved, then the iteration terminates, so these values have to be set sufficiently high. The same inputs from **line #5**, **line #6,** and **line #11** are required for all of the **var** options.

2. **var** = 'sc'. The scale factors that are normally given in **line #4** will be iterated. With this option, the scale factors that are given in **line #4** will be read by **SMERF** but then ignored. This option automates the computation of the critical diameter of an explosive, for example. Another example of its use is to determine the threshold size of a fragment impacting on an explosive. It is assumed in the iteration that a large value for the scale factor will lead to a detonation, while a small value will not.

3. **var** = 'vy'. The initial vertical velocity of selected materials will be iterated. The initial material geometry is given in the matcor file described later. Included in the geometry description is the initial velocity of each material region. When **var** = 'vy', each non-zero initial vertical velocity is set to the same current value of the iterated parameter. This option is used to find the threshold velocity for detonation for fragment or flyer impact on explosive targets.

4. **var** = 'vx'. The initial horizontal (or radial) velocity of selected materials will be iterated. This option is the same as the vertical velocity option above.

Line #3. If there is to be a restart from a previous run, the input from line 3 will assign the value **true** to **restart** and the name of the restart file to **filnam**. Restart files are dumps of the COMMON block made in previous calculations. The frequency of dumps during a run is controlled by the **outdata** file described in Section D, *Output Control*. When the restart option is set, two other files (in addition to the six basic input files) containing history data are needed. The first is **prestart,** which contains the maximum pressure of each material as a function of time. It can be obtained by copying **pmatout** which is generated by each **SMERF** run. Second, if the **lgauge** option is **true** (line 9 below), then the file **grestart** is required. It is obtained by copying the file **gaugeout** that results from each **SMERF** run whenever **lgauge=true**. It should be noted that the material properties can be changed prior to a restart, provided that the material property that is changed does not affect the mass or energy of the material. This is done by editing the equation of state data file (**eosfilname** as input in line 17). It should also be noted that the restart option can be used when the **SMERF** code is in the iteration mode (line 2 above) when **var** = 'p1', and the restart file was created prior to involvement of the acceptor explosive in the event.

Line #4. Scale factors that multiply all input spatial dimensions (**xscale**) and all input time quantities (**tscale**) are set in this line. The scale factors can greatly simplify the input for SMERF. For example, the length units for **SMERF** are centimeters, but it is sometimes convenient to provide input in inches and then set the **xscale** to 2.54. Sometimes one may want to run a sequence of problems where the only change is the size of the objects involved. It is easy to convert the input for one size problem to another by changing the **xscale** factor. In this case, one should also change the time scale factor, **tscale,** by the same amount.

Line #5. The problem stop time (**tmax**), stop cycle (**maxcyl**), and maximum Courant number (**safty**) are input in line 5. The Courant number governs the size of the integration time step. If the number is too large, then the calculation can become unstable, and it will terminate. If the Courant number is too small, then the computer may become excessive. The value of **safty** can range from about 0.4 to 0.5. Sometimes difficulties will be obtained with some aspect of the calculation, and the program will terminate. A solution might be to repeat the calculation with a smaller value of **safty**.

Line #6. Information in line 6 allows execution to terminate early, depending on the reaction of the explosive. **lpexit = true** stops computation if detonation is detected or if it is determined that no detonation will occur. Detonation is assumed if the pressure in the explosive, as designated by the material number **iffburn** described below, exceeds **pexit**. The program keeps track of the maximum pressure in the explosive. Detonation failure is assumed when the explosive pressure falls by a factor **pratio**. When detonation or failure is detected, the program terminates **ncount** cycles later. A value of **ncount** greater than zero is sometimes useful to get a final plot showing development of the detonation. If **lpexit = true,** then **SMERF** will terminate in one of two modes. For **modestop**=0, the calculation stops if detonation is detected. If **modestop**=1, the run stops if the detonation is produced and then fails to propagate in the x direction to **xstop** and in the y direction to **ystop**. If **xstop** lies outside of the mesh, then it is ignored. However, if **ystop** initially lies above the mesh, the mesh may eventually be shifted upward (see line 7) to include **ystop**. At this point, **ystop** becomes effective.

Line #7. If **lslide = true** in this line, then the grid will be moved upward to follow material motion such as shock waves or moving projectiles. If motion is detected at the top of the grid, then another row of elements will be added to the top, and the bottom row will be deleted. This can save enormous amounts of computer memory and time for calculation of projectile impact onto semi-infinite explosives, explosive critical diameter experiments, or self-forging fragment simulation. Often it is desired to simulate impacts with targets that lie outside the mesh. This sliding option can be used to incorporate a target with material number **ntarget** lying between vertical coordinates **ytmin** and **ytmax**. Whenever this option is used, the mesh (as specified in the **mshgen** data file) must have uniform spacing in the vertical direction.

Line #8. The coordinate system is specified in line 8 as Cartesian (**igeom=0**) or cylindrical (**igeom=1**).

Line #9. Boundary conditions along the left, right, bottom, and top (**b1,b2,b3,b4**) of the grid are set in line 9 and can be either reflective (**0**) or transmissive (**1**). The transmissive boundary allows disturbances and material to pass out of the computational domain, while the reflective boundary behaves as a perfectly reflecting plane.

Line #10. If a programmed burn is desired, then **lpburn=true** in line 10. The remaining three parameters in line 10 indicate the material number of the material to be program burned (**ipgburn**) and the point of initiation of the burn (**xdet,ydet**). The material chosen to burn must include both a solid and gas equation of state.

Line #11. If **lfburn=true** in line 11, then the Forest Fire burn is enabled for any material assigned this burn option. The Forest Fire burn is useful when one wishes to investigate the response of an explosive to a shock stimulus. The second entry in line 11, **iffburn**, is the material number of the "acceptor" explosive. This number is required when **lpexit=true** in line 6 above. In addition, when **lfburn=true,** the maximum pressure from material **iffburn** is included in the printed output from the calculation. Otherwise, the maximum pressure of all materials is printed.

Line #12. If **lgauge=true** in line 12, then the time, pressure, and speed at locations specified in input file **gaugein** are written each cycle to output file **gaugeout**.

Line #13. If material strength is to be included in the problem, **lstres=true** in line 13.

Line #14. Tensile failure of materials is allowed if **lpfail=true** in line 14.

Line #15. Line 15 contains switches for plotting data. When **lplot=true** and the simulation time exceeds **tplot**, the data for pressure contour plots is dumped every **dtplot** microseconds into directory **splotdir.**

Line #16. The integer **ieos** is a variable that is read but not used by the **SMERF** program. The file name for the appropriate equation-of-state data package is **eosfil**.

Line #17. The last line of **setup** contains the equation-of-state numbers of the materials in the problem. These numbers identify materials found in the equation-of-state data file **eosfilnam**.

B. THE MESH (The *mshgen* File)

The computational mesh is generated using data in input file **mshgen** which supplies the coordinates for the entire mesh (x_1,x_2,y_1,y_2) and a finely zoned subgrid (xsg_1,xsg_2,ysg_1,ysg_2) with constant cell size (**dxs,dys**). Zones grow outwardly from the subgrid by the factors **xfac,yfac**. The number of materials in the problem (**nmat**) is input in the first line of **mshgen**. The example problem discussed in Section IV uses a simple mesh with constant 0.2-cm square cells throughout. The **mshgen** file for it looks like the one shown in Table 3.

TABLE 3. mshgen Input File for the Bomb Stack Example Problem.

```
6                [nmat]
0.00,  45.00     [x1 , x2]
0.00,  45.00     [xsg1 , xsg2]
0.20,   1.00     [dxs , xfac]
0.00,  45.00     [y1 , y2]
0.00,  45.00     [ysg1 , ysg2]
0.20,   1.00     [dys , yfac]
```

C. MATERIAL ZONING (The *matcor* File)

The last line in **setup** contains the equation-of-state numbers of the different materials in the problem. These numbers are used to assign equation-of-state data to the materials. The location of each material in the mesh is determined by the data supplied in the file **matcor**. This geometry data is in the form of a sequence of ordered points and arcs. The data read in **matcor** specifies the boundary of a material body. The data is in the form of an integer (**itype**) followed by a coordinate pair defining a point. The options available for **itype** are listed in Table 4 and illustrated in Figure 5. For some values of **itype**, the coordinate represents a vertex on a polygon. For other values of **itype**, the coordinates represent the center of an arc. In this case, the counterclockwise arc length in degrees is then read by **SMERF** in the following line. The first entry must have **itype**=1, and the coordinate pair then represents a starting point on the material boundary. The sequence is terminated by the coordinate pair (**nexit**, 0.0), where **nexit** is a number greater than 1.0e+23.

The next line in **matcor** contains the initial state for the material object that was just defined. The six entries on this line give the material number, starting pressure (megabars), temperature (degrees Kelvin), horizontal velocity (cm/μs), vertical velocity (cm/μs), and history variable. If the material has a gas equation of state, then the history variable is the mass fraction of the explosive gas products. If the material is porous, then the history variable is the initial porous volume divided by the specific volume of the solid matrix.

Sometimes a material object will be used repeatedly at different places on the mesh or it may be more convenient to describe it at one location and then rotate and move it to a different location. One may define a material geometry prototype by entering **itype**=0 followed by a coordinate pair. The coordinate pair in this case represents a reference point on the prototype geometry. The material geometry boundary is then entered in the same manner as explained above until terminated by the coordinate pair (**nexit**, 0.0). The prototype geometry data is saved in a buffer. At any point during the **matcor** data input, the prototype geometry can be used to construct a material object on the mesh. This is done by entering a line with **itype**=-1 and a destination coordinate pair followed by an angle on the next line. The prototype geometry region is rotated in the counterclockwise direction about the reference point, and then the entire object is translated until the reference point coincides with the destination point. A material properties line is then read in the usual format. The prototype geometry can be used as many times as desired to form many material objects. Each material object can have different materials and different initial conditions. Only one prototype can exist at a time. If data for a second prototype geometry is entered, it replaces the previous one.

TABLE 4. Options for Connecting the Points Specified in File matcor.

itype	point	relative to	last point
1	(x,y)	(0,0)	line
2	(r,θ)	(0,0)	line
3	(x_c,y_c)	(0,0)	arc
11	(x,y)	last pt.	line
12	(r,θ)	last pt.	line
13	(r_c,θ_c)	last pt.	arc

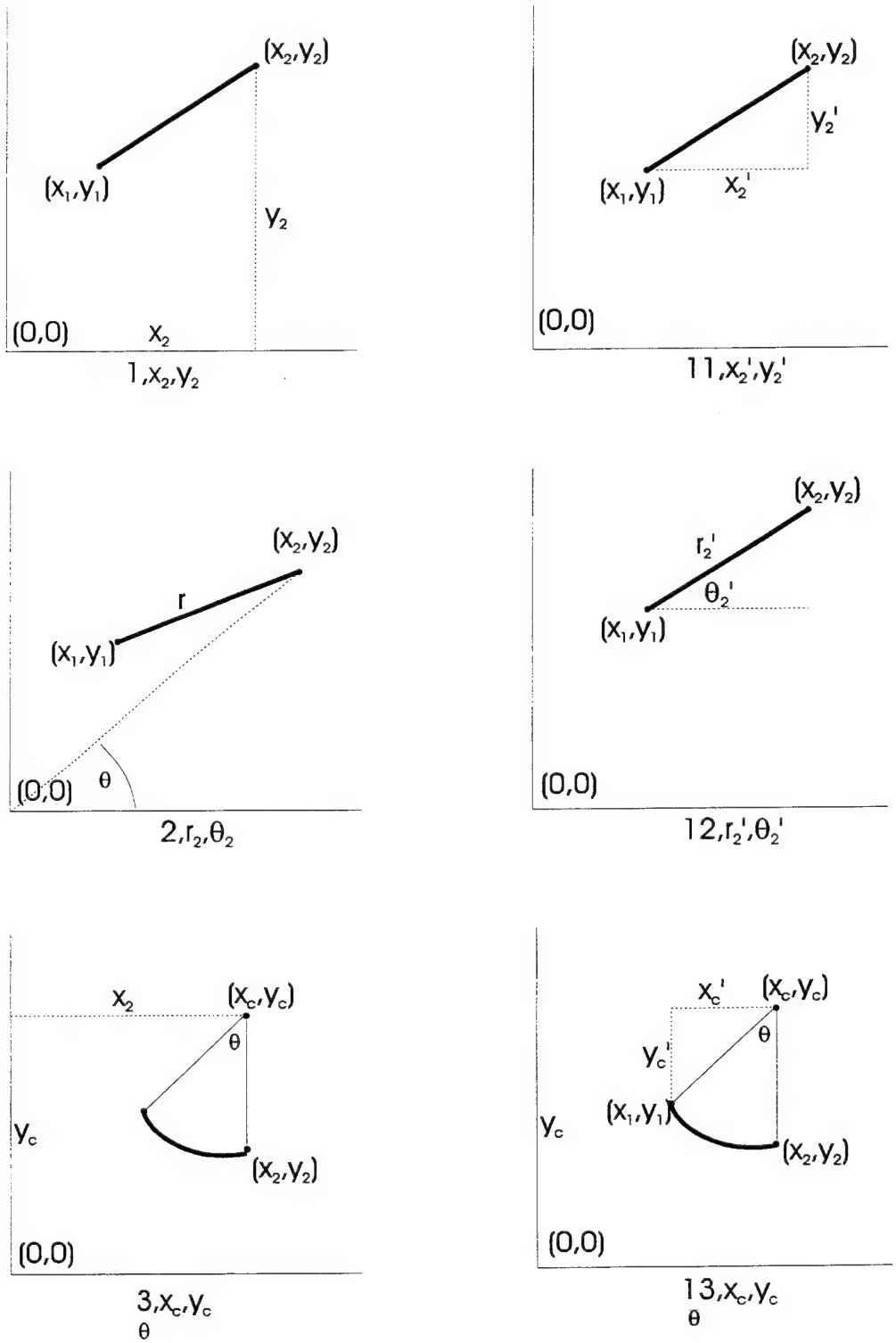


Figure 5. Different Ways to Connect Two Points in the SMERF Setup. Material regions are defined on the mesh by using one or more of these options to generate geometrical shapes.

The **matcor** file used in the bomb stack example problem in Section IV is shown in Table 5. This example uses the prototype geometry option. A prototype bomb case is defined at the coordinate origin, for example, and then it is moved to construct the donor and three acceptor bomb cases.

TABLE 5. matcor Input File for the Bomb Stack Example Problem.

```
1,      -10000. , -10000.
1,      -10000., +10000.
1,      +10000., -10000.
1,      +10000., -10000.
1,      5.e+23, 0.
6, 1.0e-6, 300., 0.0, 0.0, 0.0        Air
-1, 0.0, 0.0
1, 0.0, 13.65
3, 0.0, 0.0
360.
1, 5.0e+23, 0.                        Case Prototype
0, 0.0, 0.0  0.0
1, 1.0e-6, 300., 0.0, 0.0, 0.0        Case 1,1
0, 28.57, 0.0
0.0
2, 1.0e-6, 300. 0.0 , 0.0, 0.0        Case 1,2
0, 0.0, 28.57
0.0
2, 1.0e-6, 300., 0.0 , 0.0, 0.0       Case 2,1
0, 28.57, 28.57
0.0
3, 1.0e-6, 300., 0.0, 0.0, 0.0        Case 2,2
-1, 0.0 , 0.0
1, 0.0 , 12.65
3, 0.0 , 0.0
360.
1, 5.0e+23, 0.                        HE Prototype
0, 0.0 , 0.0
0.0
4, 1.0e-6, 300., 0.0 , 0.0 , 0.0      HE 1,1
0, 28.57 , 0.0
0.0
5, 1.0e-6 , 300. , 0.0 , 0.0 , 0.0    HE 1,2
0, 0.0 , 28.57
0.0
5, 1.0e-6, 300. , 0.0 , 0.0 , 0.0     HE 2,1
0, 28.57 , 28.57
0.0
5, 1.0e-6 , 300. , 0.0 , 0.0 , 0.0    HE 2,2
```

There are several important rules concerning input of materials into the mesh:

Rule #1. If material regions overlap, the last region specified is used. For example, a rectangular steel plate with a circular hole would be constructed by first specifying the steel rectangle followed by a circle of air.

Rule #2. Every computational zone must contain a material. Air is often the best material to use to fill the mesh and play the part of vacuum.

Rule #3. Air must be the last material number as input in data file **setup**. Air must be included in all problems. It must be the last material.

Rule #4. Material regions can extend beyond the mesh boundary. **SMERF** truncates the regions to fit within the mesh.

Rule #5. Material regions (polygons) are automatically closed. For example, a triangle can be specified by three entries.

D. OUTPUT CONTROL (The *outdata* and *gaugein* Files)

Three types of output are available from **SMERF**: (1) binary COMMON dumps, which can be used for restarts or as input to graphics packages; (2) material maps written to the screen; (3) contour plots; and (4) Lagrangian time histories for pressure and particle speed. The timing of the COMMON dumps, material maps, and contour data is set in **outdata** where a 'D' or 'd' (dump COMMON), an 'M' or 'm' (material map), or a 'P' or 'p' (contour plot) followed by the output time is specified. The material map is a *nx* x *ny* matrix of symbols indicating the location of the various materials in the computational mesh at a given time. This map is a simple way to view the state of the calculation as it is running. It is written to standard output. For the example problem discussed in Section IV, the file **outdata** looks like the example in Table 6.

TABLE 6. outdata File for the Bomb Stack Example Problem.

```
'M' ,  0.0
'M' ,  1.5
'D' ,  2.0
'M' ,  2.5
'P' ,  3.0
```

The Lagrangian history of the flow is often required. The initial locations of the tracer particles are specified in **gaugein**. The first entry is the number of gauge points, with coordinates on the lines following. A simple example file containing two points is given below.

TABLE 7. A gaugein File With Two Lagrangian History Points.

```
2
0.000 , 5.000
2.335 , 8.990
```

E. EQUATION OF STATE CONSTANTS (The *eosdata* File)

Equation-of-state constants for materials are found in an equation-of-state data file. The library supplied with the code is called **eosdata**. The first line for each material contains two entries, the equation-of-state number assigned to that material and the name of the material. The second line contains five entries which assign specific models to the material: **nmat** (material strength), **neoss** (solid/reactant eos), **neosg** (gas/product eos), **nburn** (high explosive burn), **npor** (porosity). The solid/reactant eos category designates the reactant or original state of the material (whether it is in solid or gas form), thus the inclusion of the "perfect gas" eos in this category. The gas/product eos category is reserved for the equations of state of the products of chemical reaction in explosive materials.

The options for the constitutive model categories are described in Table 8 (in reality, a group of four tables). Inspection of the table shows that the units of the input parameters are not entirely consistent. In the early development of the SMERF equation-of-state subroutines, material data was obtained from a variety of databases which sometimes used different units. In particular, the databases used were for the DYNA2D code from the Lawrence Livermore National Laboratory and the 2DE code from the Los Alamos National Laboratory. Not all of the features from these codes were implemented so that some of the constants in the present database are not used. The entries in **eosdata** for the materials used in the example problems are listed in Table 9.

TABLE 8. Definition of the Equation of State Constants Read in the eosdata File.

NMAT	MODEL/EQNS	SYMBOL	NAME	UNITS/COMMENTS
0	None			Only for perfect gas
9	Fluid	ρ_o	Initial Density	g/cm^3
		P_c	Spall Pressure	Mb (must be negative)
10	Elastic -	ρ_o	Initial Density	g/cm^3
	Perfectly Plastic	μ	Shear Modulus	Mb
		Y	Yield Strength	Mb
		E	Hardening Modulus	not used
		P_c	Spall Pressure	Mb (must be negative)
		A_1	Yield Constant	not used
		A_2	Yield Constant	not used
		P_s	Yield Constant	not used
NEOSS	**MODEL/EQNS**	**SYMBOL**	**NAME**	**UNITS/COMMENTS**
2	Perfect Gas	γ	Ratio of Sp. Heats	dimensionless
	Eq. C1.2	Q	Energy	Mb
4	Gruneisen	C	Hugoniot	cm/μs
	Eq. C1.3	S1,S2,S3	Coefficients	
		γ	Gruneisen Coef.	
		S_a	Vol Dependence of	dimensionless
120	HOM	A,B,C,D,E	Coefs. in Eq. C1.5c	
	Eq. C1.5	K,L,M,N,O	Coefs. in Eq. C1.5e	
		Q,R,S,T,U	Coefs. in Eq. C1.5d	
		C_v	Specific Heat	cal/g
		Z	Energy Offset	Mbar- cm^3/g

TABLE 8 (Cont'd). Definition of the Equation of State Constants Read in the eosdata File.

NMAT	MODEL/EQNS	SYMBOL	NAME	UNITS/COMMENTS
3	Arrhenius	E	Activation Energy	kcal
		Z	Frequency Factor	sec^{-1}
7	Ignition & Growth	$F_{mxig}, G_o, C_r, n1$	Coefs in Eq. D3.2	
	Eqs. D3.2-D3.4	$F_{mxgr}, G_1, \delta_1, r_1, m$	Coefs in Eq. D3.3	
		$F_{mngr}, G_2, \delta_2, r_2, n$	Coefs in Eq. D3.4	
11	Forest Fire	n		not used
	Eq. D4.1	P_1	Pop-Plot Point	Kb - note the units!
		-S	Pop-Plot Slope	must be negative
		P_2		not used
6	JWL	A , B	Coefficients	Mb
	Eq. C1.6	R_1 , R_2	Exponents	dimensionless
		wC_v	Gruneisen x Sp Heat	Mb- cm^3/g-K
		C_v	Specific Heat	Mb- cm^3/g-K
		E_o	Energy	Mb- cm^3/g
100	HOM	C_1 , S_1	Hugoniot $U_s=C+SU_p$	C is cm/ms
	Eq. C1.4	V_m , C_2 , S_2		not used
		γ	Gruneisen Coef.	dimensionless
		C_v	Specific Heat	cal/g
		V0		not used
		α	Thermal Expansion	cm/K
		P_s , U_s , T_o		not used
NPOR	**MODEL/EQNS**	**SYMBOL**	**NAME**	**UNITS/COMMENTS**
0	None			
3	P-α	n_s		
		ρ_p	Initial Density	g/cm^3
		a_p	Elastic Sound Speed	cm/μs
		(Vci,Pci,i=1,n_s)	Compaction Curve	cm^3/gm , Mb

TABLE 9. Entries in eosdata for Materials Used in the Bomb Stack Example Problem.

```
1, 'AIR'
1, 2, 0, 0, 0,
1.4, 1.0e-5, 0.0,

20, 'AFX-1100 ',
10, 110, 6, 11, 0,
1.492, .0500, .001, 0.0, -0.01,
0, 0, 1,
.206, 2.16, .01, 0.00, 0.00,
1.50, .259, .670, 5.0e-5, 0.0,
0.0, 300.0,
3.91, 0.0252, 4.71, 1.18, 2.0e-6, 1e-5, 0.054,
6, 84.4, -2.216, 0.0,    [ P1 = 84.4kbar ]

104, 'HOM STEEL',
10,110 ,0,0,0,
7.9, 0.769, .010, 0.0, -0.020,
0, 0, 1,
4.580E-01, 1.510E-00, 6.500E-02,
0.000E-00, 0.000E-00,
2.020E-00, 1.070E-01, 1.266E-01,
1.170E-05,1.290E-01,
1.200E-01, 3.000E+02,

117, 'HOM COMP-B',
10, 110, 120, 11, 0,
1.715, .0500, .002, 0.0, -0.01,
0, 0, 1,
.280, 1.63, .001, 0.00, 0.0,
1.5, .259, .583, 5.0E-5, 0.0,
0.0, 300,
-.35029893E+01, -.24367332E+01, .26736184E+00,
-.24236604E-01, .66789557E-03,
-.12981939E+01, .31351829E+00, .46863277E-01,
 .32738625E-02, .85793089E-04,
 .75364753E+01, -.48373016E+00, .10225829E+00,
-.15292421E-01, .76422916E-03,
 .5, .17530137,
 6, 53.9, -1.501 ,0.0,    [ P1=53.89 ]
```

F. BASIC OUTPUTS

There is no post-processor for the **SMERF code**. Certain programs have been written by the authors of **SMERF** for graphical presentation of the output, but the programs are machine-dependent and, thus, are not normally distributed. These programs read data in output files produced by **SMERF**. In this section, the contents and format of these files are explained. The names of subroutines in **SMERF** where these output files are produced are given so that the interested code user can modify them.

COMMON dumps are both written and read in subroutine ***RW***. This output is unformatted and contains essentially all the information regarding the state of the calculation at a particular time step. These files can be extremely large and are used to restart SMERF calculations as explained in the setup file description.

Because of the large size of the COMMON dumps, another means of producing data for two-dimensional contour plots is given. These output files are called **splots**. In the present version of the **SMERF** code, sufficient data is output for the production of contour plots of pressure and the history variable (burn fraction or porosity), together with material boundaries. The timing of the output of these files can be controlled through the **outdata** file using the "P" option as described above. **splots** can also be generated at regular intervals by means of input data in the **setup**, also described above. The directory where the **splots** are written is also specified in the **setup**. Names of these files are assigned according to the formula **splotxxxx** where **xxxx** is the sequence number of the files starting at 0001. They are formatted and can, therefore, be viewed and edited. Another reason for formatting the files is that the primary **SMERF** calculations are sometimes performed on a computer which is different from the one used for post-processing. Formatted files can be transferred over the network without any word length conversions. The writing of the files is done in subroutine **splot**. These files are most often used to produce animated sequences of contour plots of the simulation.

If the gauge option was enabled in the **setup,** then an output file **gaugeout** is produced. It contains formatted data representing the time histories of pressure and particle speed at specified gauge points which move with the material. This file is produced in subroutine ***GAUGE***.

The final output file is **pmatout**. It contains time histories of the maximum pressure for each material in the **SMERF** calculation. This formatted file is produced in subroutine ASTOP.

IV. EXAMPLE PROBLEMS

A. FRAGMENT IMPACT

The first example problem concerns shock initiation of an explosive from fragment impact. This is "demo" problem #1 supplied with the code on installation. The "fragment" is taken to be a steel pellet with a diameter of 1.00 cm, height of 0.20 cm, and speed of 0.12 cm/μs. The explosive is a cylinder of bare COMP-B with a diameter of 40 cm and a height of 20 cm. A schematic of the setup is shown in Figure 6.

This simulation employed a uniform rectangular subgrid (0.50 cm x 2.00 cm) embedded within a 1.00 cm x 2.00 cm computational domain. The grid cells comprising the fine scale subgrid were 0.03 cm square. These zones were allowed to expand by 3% in the radial direction for each column outside the subgrid, but the vertical cell dimension was not allowed to grow. In this way, 30 x 66 total grid cells were generated for use in the simulation. Input files for this example problem are shown in Tables 10, 11, and 12. Results of the calculation are shown in Figure 7.

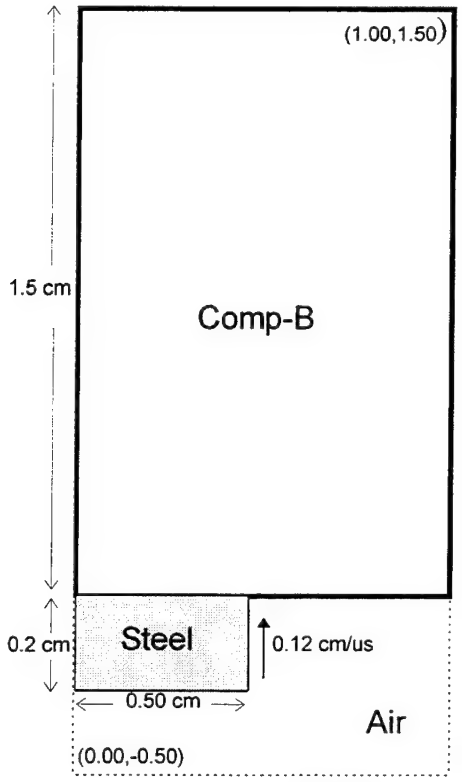


FIGURE 6. Schematic of the Setup for Shock Initiation by Fragment Impact.

TABLE 10. setup Input File for the Fragment Impact Example Problem.

'Fragment Impact'	[title]
0, 'uy', 0.1, 0.14	[niter , var , varmin , varmax]
F,'restartfile'	[restrt , filnam]
1.0 , 1.0	[xscale , tscale]
3.0, 500, 0.4	[tmax , maxcyl , safty]
T, .2, .3, 100,0,90.,90.	[lpexit , pexit , pratio, ncount, modestop , xstop , ystop]
T, 0.0, 20.0, 20.5	[lslide , ntarget , ytmin , ytmax]
1	[igeom]
0, 1, 1, 1	[b1,b2,b3,b]
F, 0, 0.0, 0.0	[lpburn , ipgburn , xdet , ydet]
T, 1	[lfburn , iffburn]
T	[lgauge]
T	[lstres]
T	[lpfail]
T, 0.0 , 0.50, '.'	[lplot , tplot, dtplot , splotdir]
2, '../eosdata'	[ieos , eosfil]
117 , 104 , 1	[material EOS numbers]

TABLE 11. mshgen Input File for the Fragment Impact Example Problem.

```
3                   [nmat]
0.00 , 1.00         [X1,X2]
0.00 , 0.50         [xsg1 , xsg2]
0.03 , 1.03         [dxs , xfac]
-0.05 , 1.50        [Y1,Y2]
-0.05 , 1.50        [ysg1 , ysg2]
0.03 , 1.00         [dys , yfac]
```

TABLE 12. matcor Input File for the Fragment Impact Example Problem.

```
1,      0.00 , -60.00
1,      25.0 , -60.00
1,      25.0 ,  60.00
1,      0.00 ,  60.00
1,      5.e+23 , 0.
3, 1.0e-6, 300., 0.0, 0.0, 0.0      AIR
1,      0.00 ,  0.00
1,      20.0 ,  0.00
1,      20.0 ,  20.00
1,      0.00 ,  20.00
1,      5.e+23 , 0.
1, 1.0e-6, 300., 0.0, 0.0, 0.0      COMP-B
1,      0.00 , -0.20
1,      0.50 , -0.20
1,      0.50 ,  0.00
1,      0.00 ,  0.00
1,      5.e+23 , 0.
2, 1.0e-6, 300., 0.0, 0.12, 0.0     STEEL
```

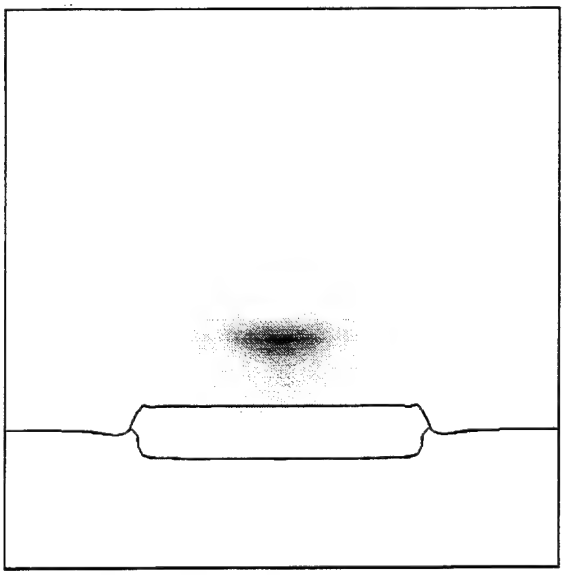


Figure 7a. Early burning in the shocked explosive at 1.0 μ s. Maximum pressure is 150 Kb.

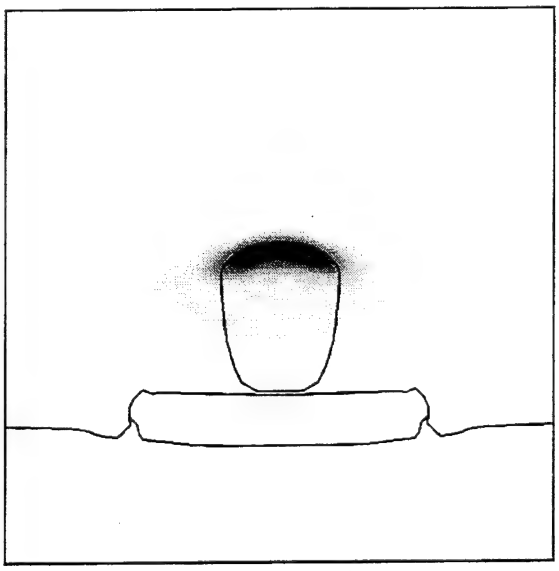


Figure 7b. Transition from shock to detonation at 1.5 μs.

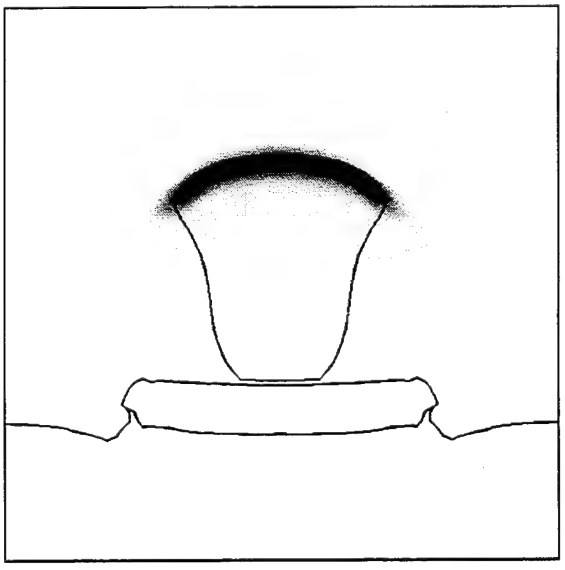


Figure 7c. Early detonation at 2.0 ms. The maximum pressure is 250 Kb.

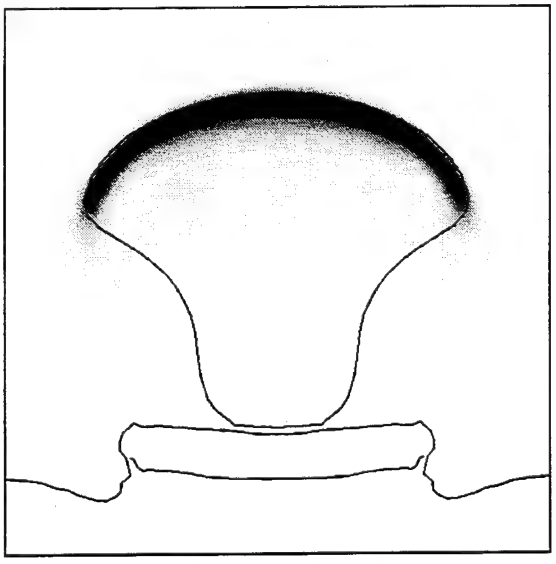


Figure 7d. Mature detonation wave in the Comp-B at 2.5 μs.

FIGURE 7. SMERF Simulation of the Shock Initiation of Composition-B by Fragment Impact. The pressure is contoured in gray scale. One set of solid black lines is the 50% reaction extent contour and the other delineates material (air-fragment-explosive) boundaries. The fragment was modeled as a steel pellet 1.0 cm in diameter and 0.2 cm thick with speed of 1.2 km/s.

B. SYMPATHETIC DETONATION IN BOMB STACKS

The second example problem is sympathetic detonation in a stack of four bombs. This is "demo" problem #4 supplied with the code on installation. The bombs are modeled as infinitely long, 13.65-cm-radius cylinders with 1.00-cm-thick steel walls. The explosive fill is AFX-1100, an experimental Air Force explosive that has been extensively characterized. Detonation of the donor bomb is modeled using the programmed burn option with detonation initiated at the center of the bomb. The Forest Fire burn option is used to simulate shock-induced reaction in the acceptor bombs. A schematic diagram of the setup is shown in Figure 8. The input files for this problem are not tabulated here. They were used as examples in Section III, which discussed code input, and can be found there.

Sympathetic detonation in a simple two-body configuration of these bombs has been tested [15]. The bombs are placed side by side and the donor bomb is detonated. With the AFX-1100 fill, the acceptor bomb was never observed to detonate, regardless of the separation distance between the two bombs. The bombs were also tested in a stack configuration. Four bombs were arranged in a rectangular stack with a 1.27-cm gap between adjoining bombs. It was found experimentally that sympathetic detonation occurred in the acceptor bomb located diagonally from the donor bomb. Figures 9a through 9d show pressure contours in the simulated bomb stack at 4 different times. The first figure shows programmed detonation in the donor bomb. The second figure shows the donor case accelerating toward the diagonal weapon. The third figure shows the onset of burning in the diagonal acceptor. Sympathetic detonation is clearly seen in the last figure.

In this calculation it is important to note that transition to detonation is a natural result of the combustion of the explosive which is modeled solely on the basis of independent small-scale tests. There are no additional switches that turn detonation off or on. The result is a true prediction.

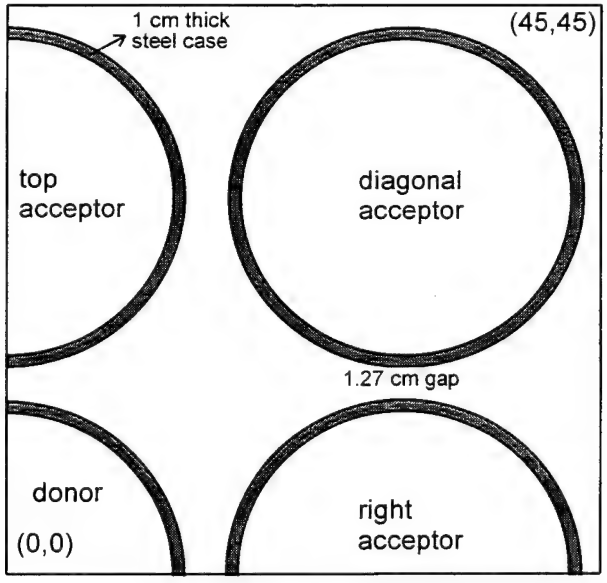


FIGURE 8. Schematic of the Setup for Simulation of Sympathetic Detonation in the Mk-82 Bomb Stack With AFX-1100 Explosive Fill.

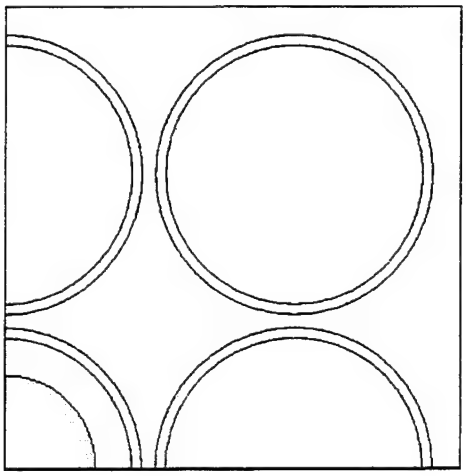


Figure 9a. Programmed burn in the donor bomb at 15 μs.

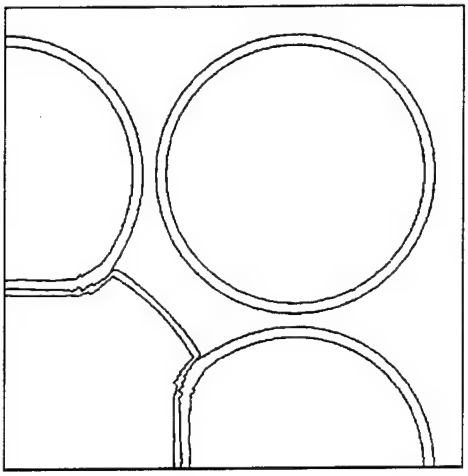


Figure 9b. Donor case accelerating towards the diagonal acceptor at 95 μs.

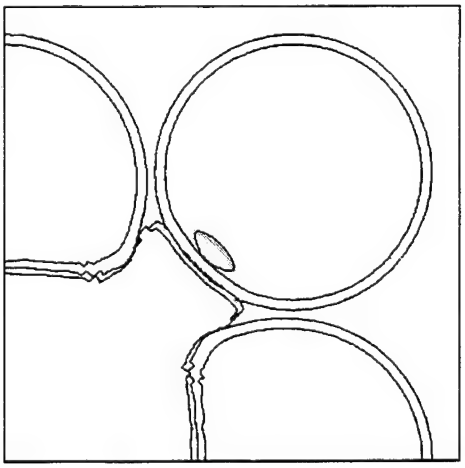


Figure 9c. Onset of burn in the addeptor due to impact shock at 135 μs.

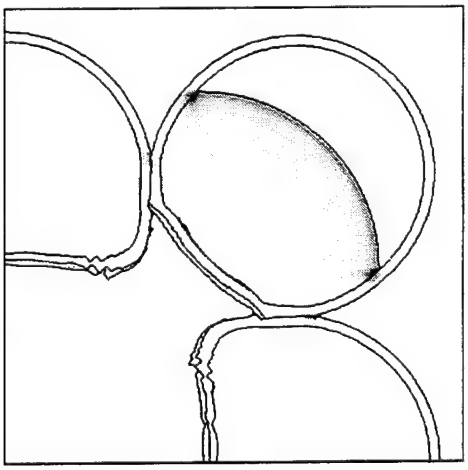


Figure 9d. Sympathetic detonation in the diagonal weapon at 160 μs.

FIGURE 9. SMERF Simulation of Sympathetic Detonation in the Mk-82 Bomb Stack. Steel case material is identified with solid black lines and the pressure field is contoured in gray scale. A 'programmed burn' was used to detonate the donor bomb and a 'Forest Fire' burn was used to compute the reaction in the three acceptor bombs. The calculation predicts detonation only in the diagonal weapon due to impact from the expanding donor case. The solid contour within the burn region in figure c is the 50% reaction extent demarkation. The explosive fill used was AFX-1100.

REFERENCES

1. Durrett, R. and D. Matuska, 1978: "The HULL Code, A Finite Difference Solution to the Equations of Continuum Mechanics," AFATL-TR-78-125, AFAL, Eglin AFB, Fl.

2. Noh,W.F. and P.R. Woodward, 1977: SLIC (Simple Line Interface Calculation), Lecture Notes in Physics No. 59, p. 330, Springer-Verlag, Berlin.

3. van Leer, B.,1977, "Towards the Ultimate Conservative Difference Scheme. IV. A New Approach to Numerical Convection," J. Comput. Phys., 23, 276-299.

4. Mader, C.M. 1979: "Numerical Modeling of Detonations", Univ. of Calif. Press.

5. Lee, E.L., H.C. Hornig, and J.W. Kury, 1968, "Adiabatic Expansion of High Explosive Detonation Products," LLNL, Livermore, CA. UCRL-50422.

6. Lee, E. L, and C. M. Tarver 1980: "Phenomenological Model of Shock Initiation on Heterogeneous Explosives", Phys. Fluids, 23, 2362.

7. Tarver, C. M. and J. O. Hallquist 1981: "Modeling Two-Dimensional Shock Initiation and Detonation Wave Phenomena in PBX-9404 and LX-17", NSWC MP-82-344, 7th International Symp. on Detonation, Naval Surface Weapons Center, Dahlgren, VA.

8. Mader, C. L. and C. A. Forest 1976: "Two-Dimensional Homogeneous and Heterogeneous Detonation Wave Propagation", LA-6259, LANL, NM.

9. Lundstrom, E.A. 1988: "Evaluation of Forest Fire Burn Model of Reaction Kinetics of Heterogeneous Explosives, NWC TP-6898, Naval Weapons Center, China Lake, CA.

10. Ramsay, J.B. and A. Popaloto, 1965, "Analysis of Shock Wave and Initiation Data for Solid Explosives," Fourth International Symposium on Detonation, ACR-126, 233.

11. Herrmann, W. 1969, "Constitutive Equation for the Dynamic Compaction of Ductile Porous Materials", J. Applied Phys., 40, 2490.

12. Kerley, G., 1992, "CTH Equation of State Package: Porosity and Reactive Burn Models", Sandia National Laboratories Report SAND92-0553.

13. Shames, H.I. and F.A. Cozzarelli, Elastic and Non-Elastic Stress Analysis, Prentice Hall, Englewood Cliffs, NJ.

14. Herrmann, W., 1991, Proceedings, 3rd Int. Conf. on Constitutive Laws for Eng. Materials, Tucson, AZ, 1991.

15. Glenn, J. G., Aubert, S., McCormick, M., and Gunger, M. E., 1993, "Sympathetic Detonation Predictive Methods", WL-TR-7001, Wright Laboratory, Armament Directorate, Eglin AFB, FL.

INITIAL DISTRIBUTION

19 Naval Air Systems Command, Arlington
- AIR-4.0T3, Magnelli (1)
- AIR-4.1 (1)
- AIR-4.1.4 (1)
- AIR-4.1.4.1 (1)
- AIR-4.1.5 (1)
- AIR-4.1.10, Barber (1)
- AIR-8.0H (1)
- ET-12 (1)
- PMA-201 (5)
- PMA-242 (1)
- PMA-253 (1)
- PMA-258 (1)
- PMA-259 (1)
- PMA-268 (1)
- PMA-280 (1)

3 Chief of Naval Operations
- OP-410 (1)
- OP-411 (1)
- OP-982E4 (1)

1 Chief of Naval Research, Arlington (OCNR-410)

7 Naval Sea Systems Command, Arlington
- SEA-665, E. Kratovil (1)
- SEA-6652, P. Wright (1)
- SEA-91 (1)
- SEA-91WM, Dr. R. Bowen (4)

1 Naval Air Warfare Center Aircraft Division, Lakehurst (Code 53, Systems Engineering and Standardization Department)

1 Naval Air Warfare Center Weapons Division, Point Mugu (Code P03512, Morcos)

1 Naval Air Warfare Center Weapons Division Detachment, Kirtland Air Force Base (Code 22, J. Stickney)

3 Naval Construction Battalion Center, Port Hueneme
- Code L51
  - W. Armstrong (1)
  - W. Kennan (1)
  - J. Tancreto (1)

3 Naval Explosive Ordnance Disposal Technology Center, Indian Head
- Code 044, R. Poe (1)
- Code 20 (1)
- Code 60 (1)

11 Naval Surface Warfare Center Division, Crane
- Code 4021, D. Wildridge (1)
- Code 4025, L. Wilson (1)
- Code 4041
  - Cooper (1)
  - Papp (1)
  - Whorrall (1)
- Code 4071
  - Broxton (1)
  - Lohkamp (1)
- Code 50 (1)
- Code 5011 (1)
- Code 50223 (1)
- Technical Library (1)

17 Naval Surface Warfare Center Dahlgren Division, Dahlgren
- G06, R. Stayton
- G20, D. Brunson (1)
- G22
  - Grigsby (1)
  - Holt (1)
  - Hoye (1)
  - Swierk (1)
  - Vittoria (1)
  - Waggener (1)
  - Wilson (1)
- G31, Monolo (1)
- H11 (1)
- H12 (1)
- R15
  - D. Houchins (1)
  - J. Powell (1)
- R35, Dr. B. Smith (1)
- G. Credle (1)
- D. Dickenson (1)

16 Naval Surface Warfare Center, Dahlgren Division, White Oak Detachment, Silver Spring
- G08
  - E. P. Johnson (1)
  - R. Kavetsky (1)
- G40
  - J. Bagnall (1)
  - D. Grenier (1)
- G81, O. Parrent (1)
- G411, A. Munach (1)
- R10H, Swisdak (1)
- R101, Roslund (1)
- R12, Spahn (1)
- R13, P. Miller (1)
- R16
  - Baudler (1)
  - Mensi (1)
- Technical Library (1)

4 Naval Surface Warfare Center Division, Indian Head Division, Indian Head
- Code 04, M. Hudson (1)
- Code 20 (1)
- Code 340, L. Miller (1)
- Code 8420G, Standardization Branch (1)

2 Naval Undersea Warfare Center Division, Keyport
- Code 04, K. Scott (1)
- C. Webb (1)

3 Naval Weapons Station Earle, Colts Neck (Packing, Handling, Storage and Transportability Center)
- Code 5011, Bender (1)
- Code 5014, Sova (1)
- Code 5021, Lee (1)

1 Office of Naval Technology, Arlington

3 Office of the Secretary of the Army
- DACS-SF, R. Fatz (1)
- DALO-SMA (1)
- JDPSS-SAF (1)

1 Army Armament, Munitions and Chemical Command, Picatinny Arsenal (AMCPM-AL, W. Liska)

1 Army Armament, Munitions and Chemical Command, Rock Island Arsenal(AMSMC-DSM-D, M. Johnson)

4 Army Materiel Command, Alexandria
- AMCAM-LG, R. Fahy (1)
- AMCDNR, Executive Director for Explosives Safety (1)
- AMCDNR-IP (1)
- AMCSF-S (1)

11 Army Missile Command, Redstone Arsenal
- AMSMI-RD (1)
- AMSMI-RD-PR (1)
- AMSMI-RD-E, D. Dreitzler (1)
- AMSMI-RD-PR-T
  - J. Murfree, Jr. (1)
  - L. B. Thorn (1)
- AMSMI-RD-SE (1)
- AMSMI-RD-SF (1)
- AMSMI-RD-ST (1)
- AMSMI-RD-TE (1)
- AMSMI-RD-TE-F, J. Porter (1)
- AMSMI-RD-TE-S, J. Knorr (1)

3 Army Test and Evaluation Command, Aberdeen Proving Ground
- AMSTE-TA (1)
- AMSTE-TC-D, H. Egbert (1)
- AMSTE-TC-M, R. Hayes (1)

22 Army Armament Research, Development and Engineering Center, Picatinny Arsenal
- AMCPM-FZ (1)
- AMSMC-QA(D) (1)
- AMSMC-QAR(D) (1)
- AMSMC-QAU(D) (1)
- AMSMC-QAW(D) (1)
- SLCHE-AR (1)
- SMCAR-AEC-TE, E. Stewart (1)
- SMCAR-AEE, M. Mezger (1)
- SMCAR-AEE-B (1)
- SMCAR-AEE-W (1)
- SMCAR-AEF-C (1)
- SMCAR-AEM, Dr. C. Westerdahl (1)
- SMCAR-AEP (1)
- SMCAR-AES (1)
- SMCAR-BAC-S (1)
- SMCAR-CC (1)
- SMCAR-ESW-F(R) (1)
- SMCAR-FMS-S, Saks (1)
- SMCAR-FS (1)
- SMCAR-FSM (1)
- SMCAR-SF, A. Larson (1)
- P. Lu (1)

3 Army Ballistic Research Laboratory, Aberdeen Proving Ground
- SLCBR-TB-B (1)
- SLCBR-TB-BB (1)
- SLCBR-TB-EE (1)

2 Army Combat Systems Test Agency, Aberdeen Proving Ground
- STECS-SOM
  - R. Gil (1)
  - R. Shipe (1)

4 Army Defense Ammunition Center and School, Savanna
  SMCAC-ES, G. Abrisz (1)
  SMCAC-ESP
    S. Blunk (1)
    J. Byrd (1)
  SMCAC-ESS, Dr. S. Kwak (1)
2 Army Engineer Waterways Experiment Station, Corps of Engineers, Vicksburg
  CEWES-SE, L. K. Davis (1)
  CEWES-SS, S. Woodson (1)
1 Army Ordnance Missile and Munitions Center and School, Fort McClellan (ATSK-CMT-Z)
1 Army Safety Center, Fort Rucker (CSSC-PR)
1 Army Technical Center for Explosive Safety, Savanna (SMCAA-ES)
2 Army White Sands Missile Range
  STEWS-TE-N (1)
  STEWS-TE-NP (1)
1 Headquarters, Dugway Proving Ground (MT-TM-A)
1 Air Force Phillips Laboratory, Kirtland Air Force Base (OLAC PL/NTE)
1 Air Force Safety Agency, Kirtland Air Force Base (AFSA/SEW,
  Chief Master Sergeant T. Beggs)
10 Air Force Wright Laboratories, Armament Directorate, Eglin Air Force Base
  ASC/YMEA (2)
    PMA-268, Crockett (1)
  ASC/SEST (1)
  ASC/YOCO(EHR), Jenus (1)
  MSD/ENS (1)
  WL/MNME
    Corley (1)
    Glenn (1)
    Dr. R. McKenney, Jr. (1)
    Parsons (1)
  WL/MNMW, Foster (1)
2 Air Force Wright Laboratories, Dynamics Directorate, Wright-Patterson Air Force Base
  WL/IGFW, W. Rieder (1)
  A. Kurtz (1)
2 Air Logistic Center, Hill Air Force Base
  ALC/MMW (1)
  TIRRPN, 6501 Range Squadron (1)
1 Defense Logistics Agency, Marrietta (K. Siler, Jr.)
1 Defense Nuclear Agency, Alexandria (Code LEEE)
1 Defense Nuclear Agency, Arlington (Code TDTR)
1 Defense Nuclear Agency (New Mexico Test Operations) Kirtland Air Force Base
  (Code TDNM-S)
2 Defense Technical Information Center, Alexandria, VA
3 Department of Defense Explosives Safety Board, Alexandria
  DDESB-IK (1)
  DDESB-KT (1)
  Dr. J. Ward (1)
1 Department of Defense, National Security Agency, Fort George G. Meade (G74(TA)
1 Deputy Assistant Secretary of Defense (FSE&S)
1 National Security Agency, Fort George G. Meade (G74(TA)
1 Office of the Assistant Secretary of Defense (FM&P/SE&S)
4 Office of the Secretary of Defense
  ADDDR&E/T&E(TFR), D. French (1)
  ADDDR&E/T&E(SP), R. Ledsma (1)
  OUSDRE(OM) (1)
  USD(A)/DDRE(R&AT/ET), R. Menz (1)
2 Lawrence Livermore National Laboratory, Livermore, CA
  L-38, Military Applications (1)
  Technical Library (1)
3 Los Alamos National Laboratory, Los Alamos, NM
  Group M7, McAfee (1)
  Group M8, Asay (1)
  Technical Library (1)
2 Sandia National Laboratories, Albuquerque, NM
  Code 7553, M. Morris (1)
  Code 9122, R. Braash (1)
1 Advanced Technology, Incorporated, Arlington, VA

2 Aerojet Propulsion Division, Sacramento, CA
  Tactical Propulsion and Munitions, H. Whitfield (1)
  Tactical Systems, D. Snyder (1)
1 Aerojet Propulsion Division, Sacramento, CA (G. Manser)
2 Alliant Techsystems, Incorporated, Brooklyn Park, NM
  Dr. K. Christianson (1)
  K. Emerson (1)
1 Alliant Techsystems, Incorporated, Ridgecrest, CA
1 Applied Ordnance Technology, Incorporated, Arlington, VA
2 Atlantic Research Corporation, Virginia Propulsion Division, Gainesville, VA
  K. Grahm (1)
  Dr. R. Snyder (1)
1 Atlantic Research Corporation, Ridgecrest, CA (R. Miller)
1 BEI Defense Systems Company, Fort Worth, TX (W. S. Marks)
1 Bobby G. Craig, Consultant, Energetic Materials, Niceville, FL
1 Boeing Aerospace Electronics, Ridgecrest, CA (G. Ebling)
1 Comarco, Incorporated, Weapons Support Division, Ridgecrest, CA (R. Dettling)
1 Denver Research Institute, University of Denver, Denver, CO
1 Fiest Engineering, Incorporated, Ridgecrest, CA
1 Hercules, Incorporated, McGregor, TX
1 IIT Research Institute, Chicago, IL (Technical Library)
1 Institute of Makers of Explosives, Washington, DC
1 Kilkeary, Scott & Associates, Incorporated, Arlington, VA
1 Lockheed Missiles & Space Company, Incorporated, Santa Cruz, CA (J. Farmer)
1 Lockheed Missiles & Space Company, Incorporated, Sunnyvale, CA (Ordnance Programs Office)
1 Martin Marietta Energy Systems, Incorporated, Oak Ridge, TN (D. Welch)
1 Martin Marietta Aerospace, Orlando, FL
1 Mason and Hanger, Amarillo, TX (A. G. Papp)
1 McDonnell Douglas Missile Systems Company, St. Louis, MO (Department E261)
1 Motorola, Incorporated, Scottsdale, AZ (Tactial Fuze Office)
1 Napadensky Engineers, Incorporated, Lutsen, MN
3 New Mexico Institute of Mining and Technology, Socorro, NM
  L. Libersky (1)
  P. McLain (1)
  P. A. Persson (1)
1 Olin Ordnance, St. Petersburg, FL
1 Olin Ordnance, Washington, DC
1 Raytheon Company, Missile Systems Division, Bedford, MA
4 Southwest Research Institute, San Antonio, TX
  P. Bowles (1)
  W. Herrera (1)
  P. Zabel (1)
  Technical Library (1)
2 Tactical Product Engineering, Propulsion Division, Sacramento, CA
  R. Brogan (1)
  J. W. Jones (1)
1 The Johns Hopkins University, Applied Physics Laboratory, Laurel, MD
1 The Johns Hopkins University, Chemical Propulsion Information Agency, Columbia, MD
1 Thiokol Corporation/Huntsville Division, Huntsville, AL (W. Thomas)
1 Thiokol Corporation, Utah Tactical Division, Brigham City, UT
1 Victor Technology, Ridgecrest, CA
1 VITRO, Washington, DC
1 Wyle Laboratories, El Segundo, CA (P. Turkheimer)
1 AWE Foulness Island, United Kingdom
1 CINO/DDEWS, Ministry of Defence, Bath, United Kingdom
2 Defense Research Establishment Valcartier, Quebec, Canada
  C. Belanger (1)
  J. F. Drolet (1)
1 DSTO/Materials Research Laboratory, Explosives Division, Victoria, Australia (M. C. Chick)
1 DSTO/Weapons Systems Research Laboratory, Ordnance Systems Division, Salisbury, Australia (L. Barrington)
1 Head of Research, Royal Ordnance Place, United Kingdom (Dr. P. R. Lee)
1 IETA Oliver Saliou, DGA/DCN Lorient, France
1 IJWA Boss van Charante, TDCK, The Netherlands

1 NATO Insensitive Munitions Information Center, Brussels, Belgium
1 RARDE/NP1, Waltham Abbey, United Kingdom
1 RARDE S/NP1 Division, Kent, United Kingdom
1 Royal Ordnance Rocket Motors Division, Worcestershire, United Kingdom
1 Sandy Smith, DRIC, Glasgow, United Kingdom